

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Diplomová práce

Nástroj pro automatickou identifikaci KIR alel

Plzeň 2020

Kateřina Kratochvílová

Místo této strany bude
zadání práce.

Prohlášení

Prohlašuji, že jsem diplomovou práci vypracovala samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 21. července 2020

Kateřina Kratochvílová

Poděkování

Ráda bych poděkovala Ing. Lucii Houdové, Ph.D. za cenné rady, věcné připomínky, trpělivost a ochotu, kterou mi v průběhu zpracování této práce věnovala. Dále bych chtěla poděkovat panu Ing. Jiřímu Fatkovi za jeho rady a pomoc při vytváření praktické části.

Abstract

The text of the abstract (in English). It contains the English translation of the thesis title and a short description of the thesis. Text abstraktu (česky). Obsahuje krátkou anotaci (cca 10 řádek) v češtině. Budete ji potřebovat i při vyplňování údajů o bakalářské práci ve STAGu. Český i anglický abstrakt by měly být na stejné stránce a měly by si obsahem co možná nejvíce odpovídat (samozřejmě není možný doslovny překlad!).

Abstrakt

Diplomová práce se zabývá identifikací KIR alel. Cílem práce je návrh a implementace nástroje pro jejich automatickou identifikaci. V práci jsou představeny KIR geny a metody získávaní genomických dat s využitím DNA sekvenace, konkrétně next-generation sequencing (NGS). Dále byly analyzovány možné/využitelné bioinformatické nástroje. Samotný identifikační nástroj byl vyvíjen na syntetických readech a nakonec testován a verifikován na datech komerčních linii DNA získaných z FN Plzeň/BC LF UK Plzeň. Vytváření syntetických readů probíhalo pomocí nástroje ART, pro zarovnávání readů na referenční DNA sekvence byl využit nástroj Bowtie2. V rámci vývoje bylo navrženo několik možných přístupů, které byly poté vyhodnoceny s ohledem na jejich možné využití.

Obsah

1	Úvod	8
2	Imunitní systém a jeho spojitost s geny	10
2.1	Geny	10
2.2	HLA a non-HLA geny	11
2.2.1	Alela a gen	13
2.3	Natural killer a jeho receptory	13
2.3.1	NKG2D receptor	14
2.3.2	KIR receptor	15
3	Sekvenační metody získávání DNA dat	21
3.1	Sanger sequencing	21
3.2	NGS next-generation sekvenování	22
3.2.1	Single-end, paired-end a mate-pair	23
3.2.2	454 sekvenování a Ion Torrent	23
3.2.3	Illumina	24
3.2.4	SOLiD	25
3.3	Metody třetí generace	25
4	Analyza dostupných bioinformatických nástrojů pro zpracování NGS dat	26
4.1	Simulační nástroje pro generování syntetických readů	26
4.1.1	ART	27
4.2	Nástroje pro zarovnávání readů	28
4.2.1	Bowtie2	29
4.2.2	Burrows-Wheeler transformace	32
4.3	Další pomocné metody	34
4.3.1	Levenshteinova vzdálenost	34
5	Implementace	35
5.1	Popis problému	35
5.2	Referenční geny	35
5.3	Testovací genotypy	36
5.4	Reálná data	36
5.5	Návrh systému	37
5.5.1	Použité programové prostředky	38

5.5.2	Bordel	38
5.6	Modulové jednotky programu	39
5.6.1	Config	39
5.6.2	Simulování dat	39
5.6.3	Zarovnání vzhledem k referenčním genům	40
5.6.4	Přístupy k identifikaci alel	40
5.6.5	Překlad alel	46
6	vyhodnocení mezi výsledku a nastavení parametru	47
7	Porovnání přístupů k identifikaci a parametrů	48
8	Závěr	49
9	Výkladový slovník pojmu a zkratek	50
Literatura		52
A	Uživatelská dokumentace	55
A.1	Spuštění programu	56
A.2	Doporučená adresářová struktura pro data	56
A.3	Výstupy programu	56
A.4	Nastavení ART a jeho spuštění	57
A.5	Nastavení ART a bowtie	57
A.5.1	pokus to nejak spustit	57
A.6	Bowtie	57
A.7	Používané soubory	58
A.7.1	FASTQ	58
A.7.2	FASTQ	59
B	Testovací genomy	60
C	Detailní výsledky	62
C.1	Experiment1	63
C.2	Experiment2	69
C.3	Experiment3	70
D	Vyhodit	79

1 Úvod

Transplantace krvetvorných buněk se využívá jako terapeutická procedura pro mnoho vážných hematologických poruch mezi které patří například akutní myeloidní leukemie. Transplantace je proces při kterém jsou dárci odebrány krvetvorné buňky které jsou následně vpraveny do těla pacienta trpícím hematologickou poruchou. Jednou z komplikací, která může nastat je reakce imunitního systému na nově vložené dárcovské buňky resp. štěp. V případě, že si štěp s imunitním systémem nebudou rozumět, může dojít k silné zánětlivé reakci, která může skončit až smrtí pacienta. V neposlední řadě může dojít také k relapsu onemocnění (návrat nemoci).

K potlačení odmítnutí dárcovského štěpu se primárně vybírají dárci podle shody v HLA znacích následovaných sekundárními znaky, které mohou být například věk či pohlaví. Shoda v HLA znacích se určuje podle shody v alelách genů HLA -A, -B, -C, -DRB1, -DQB1. Alela je konkrétní forma genu. Každý jedinec má tyto HLA geny dvakrát (jednu pětici od matky a druhou od otce) a proto se úplná shoda označuje jako 10/10. Nověji je možné se setkat s označením 12/12. Znamená to 10/10 navýšené o gen HLA -DPB1. Tento gen, ale nevyžaduje narozdíl od standarníchch HLA znaků přesnou shodu. Klíčové je zde zda patří do skupiny permisivních (tolerančních) alel, které by měli snížit možnost relapsu (návrat nemoci) a rizika transplantace. Oproti tomu některé skupiny alel naopak mohou rizika zvýšit. V poslední době se množí studie, které prokazují vliv i takzvaných non-HLA genů. Jedením z nich může být i skupina genů Killer-cell immunoglobulin-like receptor (KIR). Jednou z výhod je, že určité seskupení KIR genů snižuje riziko návratu nemoci. V případě, kdy by se rozhodovalo mezi více dárci shodných v HLA znacích, by se mohl ten vhodnější vybrat právě na základě KIR. Pro zjištění jak HLA znaků tak KIR genů se využívají sekvenační metody. [23] [12] [14]

Cílem práce je navrhnut a implementovat nástroj pro automatickou identifikaci KIR alel. Vstupní data tzv. ready jsou krátké kusy DNA (posloupnost písmen A, C, G a T) a jsou výstupem ze sekvenačních technik. Nikdo přesně neví co znamenají. Může to být gen, část genu nebo několik různých genů. Tyto ready se zarovnávají vůči referenční alelám, díky tomu je možné zjistit o které alely se jedná. Ready budou pro vývoj nástroje simulována a v konečné fázi testování budou vyměněna za data z FN Plzeň/BC LF UK Plzeň.

V poslední fázi bude vyhodnocena shoda readů a referenčních genů.

2 Imunitní systém a jeho spojitost s geny

Imunitní systém chrání organismus před škodlivinami. Skládá se ze dvou hlavních částí vrozené imunity a získané imunity. Reakce imunitního systému je vždy komplexní reakce organismu mezi jednotlivými buňkami imunitního systému reagující na přítomnost specifických antigenů. Antigeny jsou látky, které imunitní systém rozpozná a zareaguje na ně. V podstatě to může být jakákoli bílkoviná sloučenina. Antigen se obvykle nachází na povrchu buňky jako vyjádření genu. Imunitní systém následně zjistí o jaký antigen se jedná, respektivě o jakou buňku se jedná, zda tělu vlastní (např. zdravá buňka) nebo buňku tělu cizí (např. nádorová buňka), tedy jedná-li se o expresy lidského genu nebo například viru. Jedná-li se o buňku tělu cizí imunitní systém reaguje snahou ji zničit.

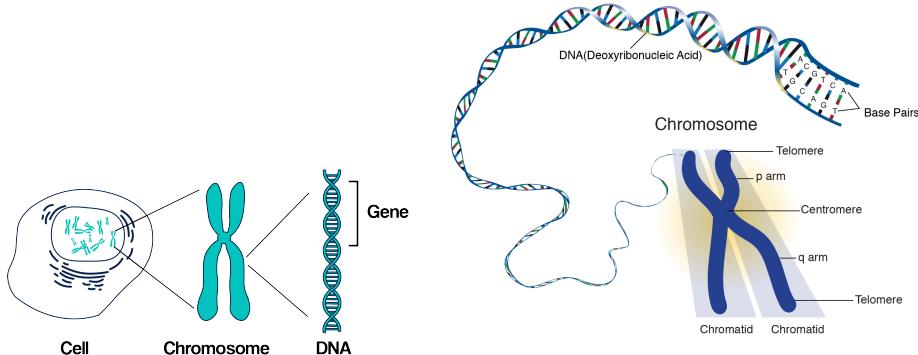
Vrozená imunita též označována přirozená, neadaptivní, antigenně ne-specifická je neměnně zapsána v DNA. To znamená, že při každém setkání s antigenem odpoví stejnou reakcí. Buňky nesoucí vrozenou imunitu jsou stálé přítomné v krvi, takže jejich případná aktivace je takřka okamžitá (minuty až hodiny). Do této imunity patří i natural killer buňky s KIR receptory, které budou dále rozebírány v textu.

Získaná imunita též označována specifická či adaptivní oproti specifické má v genomu zapsány pouze své základy. V průběhu lidského života se vyvýjí a mění. Změna může nastat například očkováním nebo proděláním patřičné choroby. Tato změna ovšem nemusí být trvalá. Z těchto důvodů může být odpověď získané imunity při setkání se stejnou chorobou rozdílná. Fungování získané imunity zajišťují T- a B- lymfocyty, ale nefunguje samostatně. Při zabíjení patogenů spolupracuje s vrozenou imunitou.

2.1 Geny

V každé buňce lidského organismu, konkrétně v buněčném jádře, je možné nalést 46 chromozomů. Jeden chromozom představuje stočenou dlouhou molekulu DNA (Deoxyribonuklenovou kyselinu). Všech 46 chromozomů obsahuje okolo 100 000 genů. Drobný segment DNA, který řídí buněčnou funkci

je právě gen. Konkrétní forma genu je alela. [30]



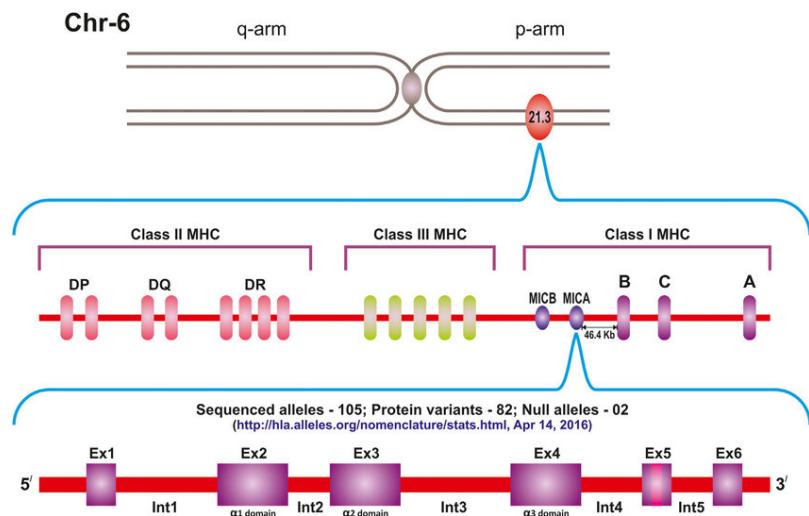
Obrázek 2.1: Převzato z [4] a [1]

Uvnitř buňky máme celý genom, který se ovšem nemusí projevit na povrchu buňky. Pokud se vlastnost, kterou gen přenáší projeví na povrchu buňky označujeme to jako exprese genu (jeho sebevyjádření). Od toho se odvíjí i konkrétní názvosloví typu gen, receptor či molekula.

2.2 HLA a non-HLA geny

Human leucocyte antigen (HLA) je genetický systém, který je primárně zodpovědný za rozeznávání vlastního od cizorodého. Někdy je termín HLA zaměňován s MHC. MHC (Major histocompatibility complex) je souhrnný termín pro všechny komplexy, kdy podskupinou jsou práve HLA (H - Human) který je pro lidi. Stejně tak existuje DLA (D - Dog) který je pro psy. Z funkčního i biologického hlediska jde však u všech savců o stejnou skupinu genů. [23]

Přesná definice mezi HLA a non-HLA geny neexistuje. Mimo jiné i jejich rozdelení není v literaturách sjednocené. Jak je vidět z obrázku 2.2 je možné geny rozdělit do tří tříd. V některých literaturách je možné nalést označení non-HLA genů jako geny III.třídy v jiné, že jsou to všechny geny III třídy a některé geny třídy I. Tato práce bude vycházet v označení gen za non-HLA či HLA z definice uvedené v [28]. Zjednodušeně tedy můžeme říci, že geny které nejsou řazeny k HLA skupinám jsou non-HLA. Je-li gen označen za non-HLA neznamená to, že by neměl souvislost s funkcí imunitního systému. Naopak má, jen ne výlučně s HLA systémem. Non-HLA geny kódují produkty spojené s imunitními procesy. Mezi non-HLA geny mimo jiné patří MICΑ, MICB a KIR. [28]



Obrázek 2.2: Šestý chromozom zobrazující HLA(-A, -B, -C, -DR, -DQ, -DP) i non-HLA(MICA, MICB) geny. Protein vzniklý expresí genu je definován exony, které definují transkripcí(přepis) do RNA. Introny při translaci(překladu) nehrají roli a často jsou sekvenovány jen exony. [8]

HLA a některé non-HLA geny se nacházejí na krátkém raménku 6 chromozomu, konkrétně 6p21.3 a zaujímá úsek přibližně jednu tisícinu genomu. Tento region je nejvíce komplexní a polymorfní na lidském genomu s více než 220 geny. Oproti tomu jedna ze skupin non-HLA genů, konkrétně KIR geny, se nachází na 19 chromozomu. Rozsáhlá diverzita genů vznikala snahou eliminovat neustále se měnící spektrum patogenů. Produkty těchto genů na povrch buňky významně ovlivňují odpověď na infekční choroby a výsledky buněčné či orgánové transplantace. [28]

Při určování shody dárce a pacienta se rozhoduje na základě shody alel u genů HLA -A, -B, -C, -DRB1, -DQB1. Díky velké diverzitě HLA genů je počet možných kombinací několik miliard. Některé kombinace genů se vyskytují na základě oblasti či národnosti častěji nebo mohou být naopak vzácné. HLA geny se obvykle dědí jako blok (celý haplotyp), avšak ve výjimečných případech může dojít k rekombinaci. Z tohoto důvodu je nejsnadnější nalést shodu v pokrevním příbuzenstvu.

Jelikož každý jedinec má dvakrát geny na pozicích HLA -A, -B, -C, -DRB1 a -DQB1 (jednu pětici od otce, druhou pětici od matky), je maximální shoda 10/10 (shoda obou alel v lokusech) popř. DPB1 a shoda 12/12. Čím je shoda menší tím větší je riziko nepřijetí stěpu. U nepříbuzných jedinců lze tolerovat shodu 9/10 či 8/10. [12] [23]

V posledních letech se objevuje Haploidentická transplantace, kdy je možné použít krvetvorné buňky příbuzného se shodou pouze jednoho haplotypu (5/10) například všichni rodiče a děti. Umožnuje to podávání chemoterapie pár dní po transplantaci, která zníčí všechny buňky, které tělo nepřijme. Využívá se toho hlavně v případech časové tísně, kdy není čas hledat dárce v registrech. [3]

2.2.1 Alela a gen

Alelu můžeme definovat jako variantu genu s nepatrnným rozdílemem v sekvenci nukleotidů DNA oproti jiné alele stejného genu. Jinak řečeno alely jsou varianty genu na molekulární úrovni. Geny se zpravidla vyskytují minimálně ve dvou formách (dvou alelách). Gen určuje výskyt nějakého znaku, například oči konkrétního živočicha budou mít barvu. Alela pak určuje jaká barva to bude. Alela zajišťuje konkrétní fenotypový projev genu.

V případě genu KIR2DL1 mohou být jeho alely 0010101 a 0010102. Zápis genů tak, jak s nimi budeme pracovat může vypadat způsobem zobrazeným v 2.2.1.

```
> KIR : KIR00001 KIR2DL1 * 0010101 14738 bp (2.2.1)
GTTCCGGGAGGTTGGATCTCAGACGTG...
```

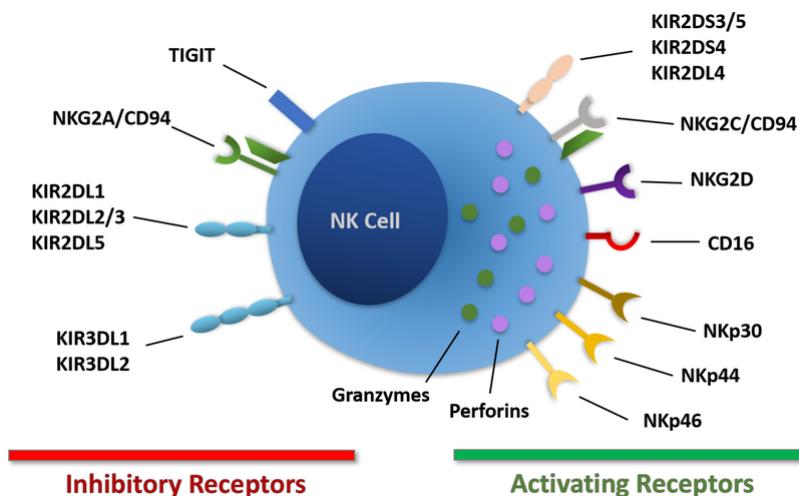
Označení *KIR : KIR0001* označuje pořadové číslo, kdy alela byla nalezena. Oproti tomu *KIR2DL1 * 0010101* je označení genu podle jeho vlastnotí.

K zjištění konkrétních alelických variant se pro tzv. typizaci využívají sekvenační metody, typicky s polymerázovou řetězovou reakcí.

2.3 Natural killer a jeho receptory

Natural killer buňky (NK buňky) jsou velké granulární lymfocyty vrozeného imunitního systému. V krevním oběhu lidského těla je jich možné nalést 10–15%. Klíčovou vlastností NK buněk je nejenom schopnost rozlišit poškozené buňky od zdravých, ale i poškozené buňky rychle a efektivně likvidovat. Poškozené buňky mohou být buňky infikované virem či buňky transformované v nádorové. Na povrchu NK buňky se nachází receptory, které jsou zobrazeny na obrázku 2.3, regulující odpověď imunitního systému. Natural killer buňky oproti B- a T- lymfocytům (buňkám získané imunity) nemají

antigenně specifické receptory. Jedním ze způsobů jak NK buňky rozpoznávají a zabíjejí poškozené buňky je na základě interakce mezi KIR receptorem a HLA molekulou na povrchu zkoumané buňky (podrobněji viz sekce KIR). Stejně tak mohou zabíjet na základě receptoru NKG2D, který aktivuje cytotoxicou reakci při setkání s ligandem MICA a MICB. Ligandem označujeme malou molekulu, která se váže na vazebné místo cílového proteinu(receptoru) a vyvolává fyziologickou odpověď, která může mít inihiční či aktivační charakter.



Obrázek 2.3: Natural killer buňka a její receptory, rozděleny na aktivační a inhibiční.[9]

2.3.1 NKG2D receptor

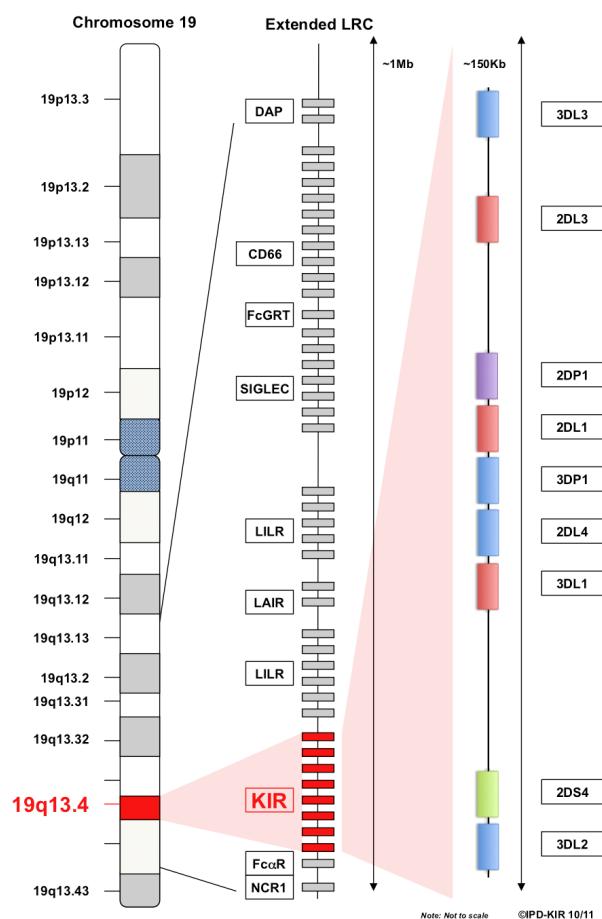
NKG2D je jeden z nejvýznamnějších aktivačních receptorů na NK buňce rozpoznávající především buněčný stres, který může spustit cytotoxicitu (shopnost níčit buňky), i když se na povrchu buňky nachází inhibiční HLA-I ligandy.

Geny skupiny MICA a MICB jsou označeny jako class I chain-related gene. To znamená, že se běžně neřadí do I. třídy MHC. Takto označované geny mají souvislost s MHC I. třídy, ale narozdíl od nich neváží peptidy. Oproti HLA genům, které mají svoje produkty na lymfocytech, se produkty MICA a MICB nachází na epitelových buňkách. Nejdá se tedy o standardní HLA geny, proto jsou nověj v literaturách označovány jako non-HLA. Jejich expresí na povrch buňky jsou ligandy, které se váží na receptor NKG2D. Buňky s ligandy MICA a MICB se množí při nádorovém onemocnění, zanětu nebo pod vlivem různých forem buněčného stresu a díky navázáním na receptor

může být spuštěna imunitní reakce. [26] [13] [9] [28]

2.3.2 KIR receptor

Killer immunoglobulin-like receptor (KIR) je skupina genů řazených mezi non-HLA geny. Jejich zvláštností je fakt, že se nenachází na 6 chromozomu, ale na 19 a tak shodní dárci HLA znaků mohou být neshodní v KIR znacích. Expresí KIR genů jsou receptory na povrchu natural killer buněk. Dnes je známo 15 genů a 2 pseudogeny rozlišujících se na inhibiční a aktivační na základě cytoplasmatického ocásku a počtu imunoglobulínových domén. [23]



Obrázek 2.4: KIR se nachází na 19 chromozomu v oblasti jménem leukocyte receptor complex (LRC). [28]

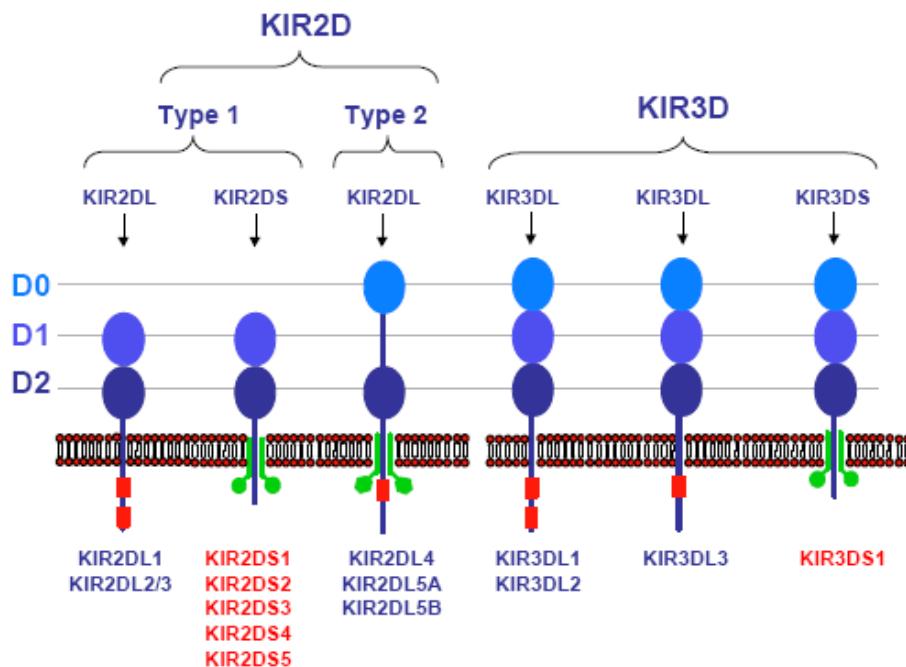
Nomenklatura KIR genů

KIR geny (na obrázku 2.5) se liší různou délkou cytoplasmatických ocásku (tail) a různým počtem imunoglobulin-like domén (lg-like). Na základě této

rozmanitosti byla založena nomenklatura KIR genů, tedy jejich pojmenování.

Jak je vidět na obrázku 2.5, cytoplasmatický ocásek může být dlouhý (long - L) nebo krátký (short - S). Je možné se setkat i s označením P, které slouží pro pseudogeny. Oproti tomu imunoglobulinové domény se mohou vyskytovat 2 (2D) nebo 3 (3D). Právě z těchto vlastností vychází základ pojmenování KIR genů.

Příkladem může být KIR2DL1*010101, kde 2D označuje dvě imunoglobulinové domény, L značí dlouhý ocásek, 1 značí že je to první 2DL protein. Numerická definice alely je poté oddělena hvězdičkou. První tři čísla označují alely, které se liší v sekvencích jejich kódovaných proteinů, další dvě číslice se používají k rozlišení alel, které se liší synonymními rozdíly v kódující sekvenci. Konečné dvě cifry rozlišují alely na základě substituce v intronu, promotoru nebo jiné nekódující oblasti. [28]



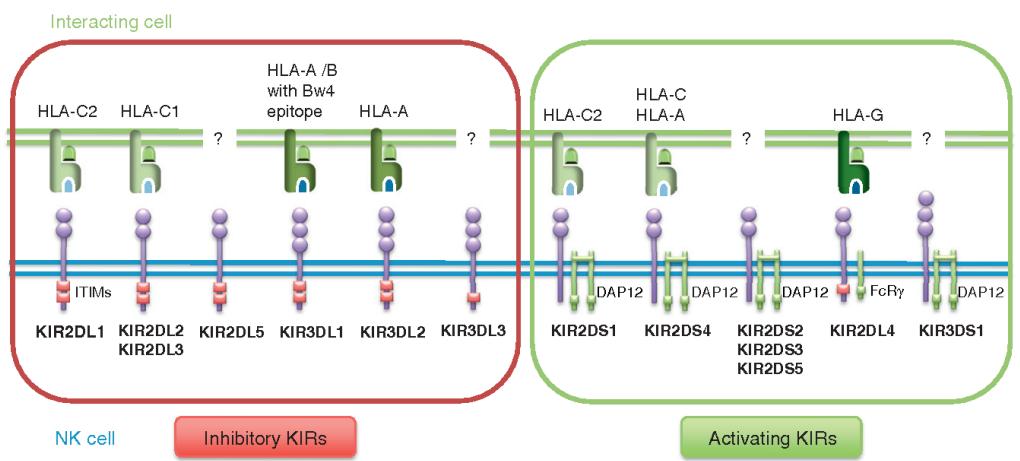
Obrázek 2.5: Nomenklatura KIR genů. [23]

Další rozdelení KIR genů je na již výše zmíněné inhibiční a aktivační. Dle obrázků 2.3 a 2.5 je možné si povšimnout detailu, že až na KIR2DL4 jsou aktivační KIR s krátkým ocáskem, zatímco inhibiční jsou s dlouhým ocás-

kem.

Aktivace NK buněk pomocí KIR

Jak již bylo výše zmíněno, KIR receptory můžeme rozdělit na inhibiční a aktivační. Zda dojde k aktivaci NK buňky rozhoduje právě jejich rovnováha na zkoumané buňce. Obrázek 2.6 uvádí vazebné ligandy pro jednotlivé KIR receptory.

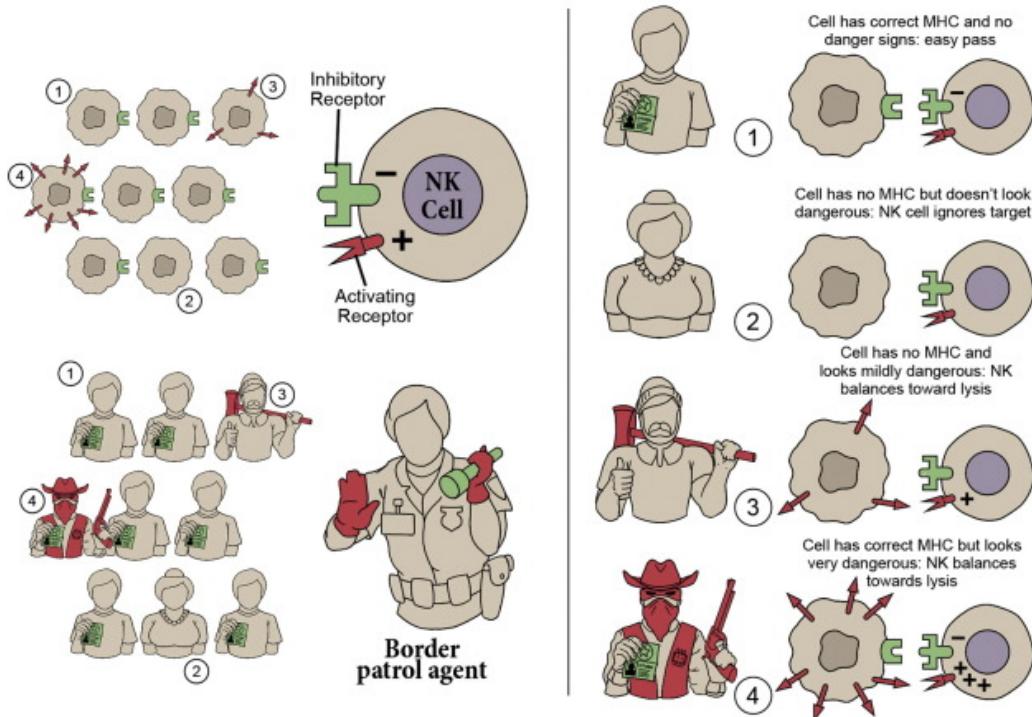


Obrázek 2.6: KIR geny a jejich vazebné ligandy. Pokud je v obrázku ?, značí to, že pro daný receptor není znám vazebný ligand. [31]

NK buňky ustavičně prohledávají své okolí a testují přítomnost příslušných HLA ligandů pro své KIR receptory. Pokud je příslušný HLA ligand přítomen, naváže se na NK buňku (2.7 případ 1). Tímto systémem jsou ochráněny vlastní buňky. Pokud přítomen není, je spuštěna cytotoxicická reakce a zkoumaná buňka je zničena.

Některé virem napadené buňky potlačují propsání HLA ligandu na povrch buňky a tím se brání cytotoxicitě proti T lymfocytům, ale tím jsou naopak více citlivější na cytotoxicitu proti NK buňkám, jak je zobrazeno na obrázku 2.7 případ 3.

The NK Cell is like a border patrol agent

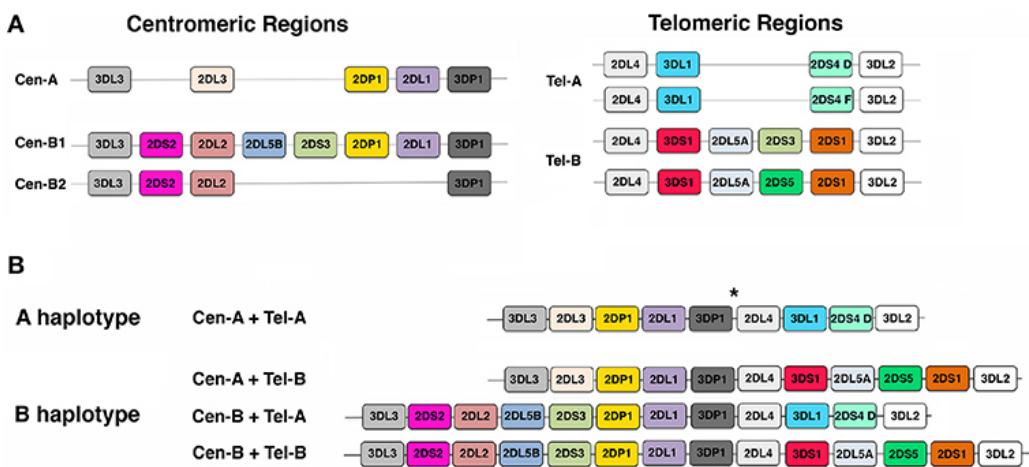


Obrázek 2.7: Přirovnání fungování natural killer buňky k pasové kontrole. V pravé části jsou zobrazené případy, které mohou nastat když natural killer buňka potká jinou buňku. V 1. případě je tělu vlastní zdravá buňka, kde se KIR receptor naváže na HLA ligand a k cytotoxické reakci nedojde. Druhým případem je červená krvinka. K reakci NK buňky opět nedojde, protože na zkoumané buňce nepřevažují aktivační receptory. V 3 případě je to nádorová buňka, která schová HLA ligand (může nastat po transplantaci kostní dřeně) a tím se "schová" proti T-lymfocytům. Avšak aktivační receptory převládají a tak k cytotoxicitě dojde. Ve 4 příkladě je nádorová buňka nebo virem nakažená buňka (stresové ligandy). Aktivační receptory převládají k cytotoxické reakci dojde.[29]

KIR genotyp a haplotyp

KIR genotyp je vyjádření, jaké konkrétní KIR geny genom obsahuje. Genotyp je možné rozdělit na dvě části, takzvané haplotypy. Jeden haplotyp je od otce, druhý je od matky. Na základě kombinací všech genů je možné vytvořit velký počet KIR genotypů. Proto byl díky shromážděným haplotypům sestaven model, který toto množství mírně redukuje, samozřejmě existují vzácné varianty, které se do tohoto modelu nehodí. Haplotyp se rozděluje na dvě části, centrometrickou a telometrickou v závislosti zda je blíže k centromere

či telomeře. Jednotlivé části mezi sebou mohou být kombinovány. Centromerická i telomerická část může být zařazena do jedné ze dvou skupin A či B na základě genů které obsahuje (viz. obrázek 2.8 část A). Celý haplotyp je následně přiřazen do jedné z dvou skupin podle kombinace centromerické a telomerické části. V případě, kdy jsou obě části A/A je haplotyp označen za A, v ostatních kombinacích (A/B, B/A, B/B) je haplotyp B (viz. obrázek 2.8 část B). Jiná definice pro rozdělení haplotypů uvádí, že skupina B musí obsahovat alespoň jeden z genů KIR2DL5, KIR2DS1, KIR2DS2, KIR2DS3, KIR2DS5 a KIR3DS1. Naopak skupina A neobsahuje ani jeden z těchto genů. Je třeba si zde uvědomit, že každý jedinec má 2 KIR haplotypy. [25]



Obrázek 2.8: Rozdělení KIR genů na centrometrickou a telometrickou část, pojmenování je na základě, zda je úsek blíže k centromeru nebo k telomeru (viz obrázek 2.1). Je možné si zde povšimnout, že je možné některé geny najít jak v centromerické části tak v telomerické části. Upraveno z [25]

Podle některých studii zabývající se vlivem KIR haplotypů na výsledky transplantace bylo zjištěno, že KIR haplotypy ovlivňují výsledky u akutní myeloidní leukémie či lymfoblastické leukemie. Ve srovnání s haplotypem A měl haplotyp B, především jeho centrometrická část, ochraný účinek před návratem nemoci a zároveň zvýšil pravděpodobnost přežití pacienta. Na základě této skutečnosti se mohou dárci řadit do tří skupin best, better a neutral. Rozřazení do třídy se využívá jako počet B a jejich umístění, v centromerické oblasti či telomerické oblasti. Mimo jiné je možné se setkat s pojmem B-skóre. Toto číslo udává počet B, které se v daném haplotypu nachází. Best je definován s B-skórem alespoň 2, přičemž dvě B se musejí

nacházet v centromerické oblasti Cen-B/B a Tel-x/x. Better je definován s B-skorém alespoň 2, aby nebyl haplotyp zařezen do Best musí být logicky alespoň jedna z Centromerických oblastí A - Cen-A/x a Tel-B/x. Neutral je v případě jedné B části nebo žádné. [10]

KIR geny se stejně jako HLA dědí celý blok. Jelikož HLA se nachází na 6 chromozomu a KIR na 19, tak shodní dárci v HLA znacích se jen menšinově shodují v KIR genech. V případě příbuzného dárce shodujícího se v HLA znacích je pouze 25% shodných také v KIR. [10]

Hapl Group	Genotype ID ¹	3DL1	2DL1	2DL3	2DS4	2DL2	2DL5	3DS1	2DS1	2DS2	2DS3	2DS5	2DL4	3DL2	3DL3	2DP1	3DP1	Populations	Individuals
AA	1																	190	7,540
Bx	2																	178	2,522
Bx	4																	178	2,096
Bx	3																	167	1,157
Bx	5																	161	1,536
Bx	6																	155	899
Bx	7																	134	583
Bx	8																	130	635
Bx	9																	120	395
Bx	71																	112	443

Obrázek 2.9: Deset nejčastější KIR haplotypů. Šedý obdélník značí přítomnost genu, bílý jeho nepřítomnost. [6]

3 Sekvenační metody získávání DNA dat

Po pojmem sekvence DNA se skrývá posloupnost písmen představujících primární strukturu reálné nebo hypotetické molekuly čí vlákna DNA, které nese nějakou informaci. Jednotlivá písmena jsou označována jako nukleotidy nebo nukleové báze. Nukleové báze mohou být A - adenin, C - cytosin, G - guanin a T - thymin. [2]

Příkladem může být následující úsek sekvence na základě obrázku 2.1

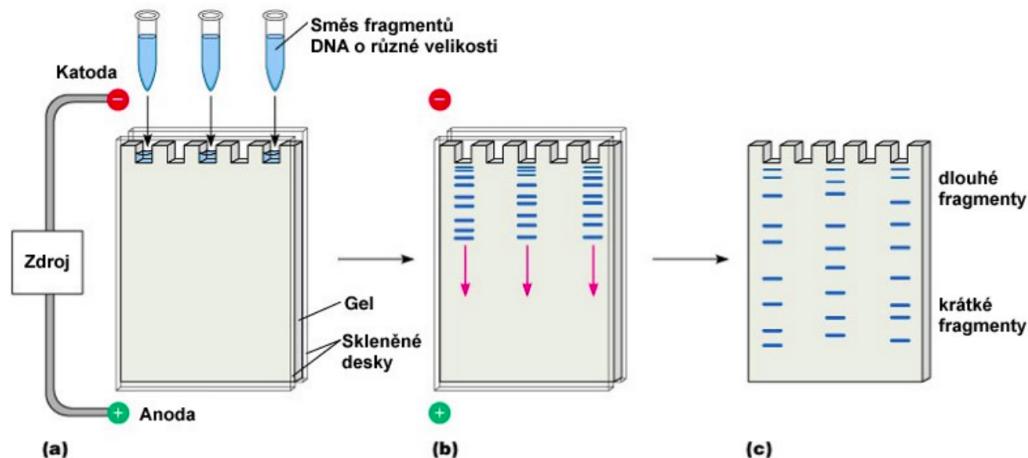
$$ACGTCA \quad (3.0.1)$$

Sekvenování DNA, někdy pouze sekvenování, jsou biochemické metody, kterými se zjištuje pořadí nukleotidů (A, C, G, T) v sekvenci DNA. Díky tomu je možné zjistit typizaci konkrétního člověka. Sekvenační metody se liší zejména délkou řetězce, kterou dokáží zpracovat, cenou a rychlostí sekvenace. Pro porovnání, sekvenování celého genomu Sangerovo metodou by stálo několik milion dolarů a trvalo zhruba 10 let. Při použití dnešních metod by cena byla zhruba tisíc dolarů. Většina sekvenačních metod využívá vlastnosti přitahováním báze do páru pouze jednou konkrétní bází. To znamená že se adenin vždy páruje s thyminem a cytosin se vždy páruje s guaninem. Z těchto párů vzniká již známá dvojitá šroubovice DNA. Při sekvenování je možná se často setkat, že se sekvenuje jen kónkrétní kus DNA, který je zrovna výzkumně čí prakticky potřeba. Největším problémem u sekvenování je, že úseky DNA vzniklé ze sekvenátoru (označovány jako ready) jsou jen kousky, které je třeba poskládat zpět. K tomu slouží zarovnávání. [17]

3.1 Sanger sequencing

Sanger sekvenování využívá možnosti namnožení řetězce díky vzájemnému přitahování konkrétních bází. V prvním kroce replikace jsou nastříhané řetězce rozdeleny na dvě vlákna. Lze si představit, že tyto dvě oddělená vlákna jsou dána do směsy, kde plavou jednotlivé nukleotydy spolu s upravenými nukletidy, které nesou specifickou fluorescenční barvu a za které není možné nic navázat. Následně za pomoci střídaní teploty volně plující nukleotidy tvoří postupné páry s řetězcem, který chceme namnožit. Pokud se povede

celý řetězec namnožit je odtržen a může se dále množit. Postupně ale bude docházet k navazováním nukleotidů s fluorescenční barvou. Tím se vytvoří nekolik různě dlouhých sekvencí zakončených označeným nukleotidem. Podle jeho barvy je možné poznat o jaký nukleotid se jedná. Následně jsou za pomocí elektroforézy seřazeny v gelu podle délky. Elektroforéza rozděluje různě dlouhé sekvence na základně odlišnosti pohybu v elektrickém poli. Kratší doputují dále než delší. Pomocí sanger metody je možné sekvenovat řetězce dlouhé až 1000 bází.



Obrázek 3.1: Elektroforéza. [24]

3.2 NGS next-generation sekvenování

Next-generation sekvenování někdy označováno jako metody druhé generace jsou v porovnání se Sangerovo sekvenováním rychlejší a levnější, na druhou stranu ale dokáží zpracovávat jen řetězce dlouhé 100 až 500 bází, mají menší přesnost a časteji chybují. Jejich rychlosť spočívá především ve schopnosti detektovat přidávání bází jednu po druhé a zároveň sekvenovat tisíce až miliony rozdílných molekul DNA najednou.

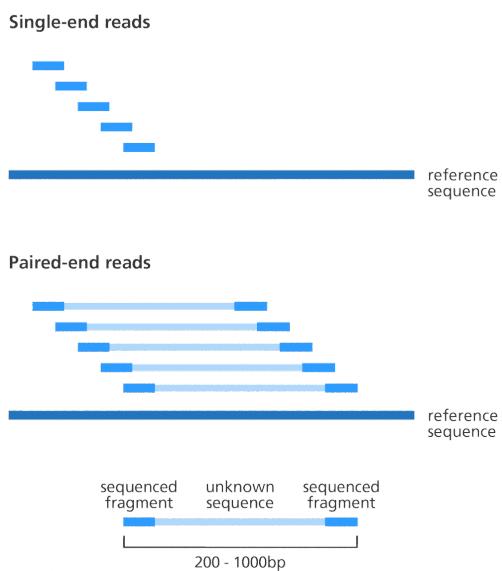
Všechny tyto metody si předpřipraví řetězce nastříháním na krátké části a připevněním takzvaného adeptéra na jejich konec. Adaptér je krátká molekula DNA, která slouží k uchycení sekvenovaného úseku na pevný povrch. Řetězce DNA jsou namnoženy díky čemuž vzniknout klastry identických molekul koncentrovaných v jednom místě. Díky tomu je posílen signál, který by z pouhé jedné molekuly nebyl dostatečně silný. Tento signál je zachycen kamerou. Jeden z důvodů popularity NGS metod jsou i cenově dostupné

stolní sekvenátory.

3.2.1 Single-end, paired-end a mate-pair

Single-end je sekvenování pouze jednoho konce molekuly. Nevýhoda tohoto způsobu se projeví především na krátkých readech, kde se zvýší problém jejich správného umístění. Oproti tomu v případě paired-end se sekvenuje z obou konci daného úseku. Vzniklé dva ready jsou označeny a zároveň je známá jejich vzdálenost, která se pohybuje od 200 do 400 bp (base pair). Mate-pair je v podstatě paired-end s rozdílem, že je mezi ready větší vzdálenosti od 2 do 5 kb (kilobase) - takže přibližně 2000 - 5000 bp. [5]

TODO obrázek je z trochu blbího zdroje nejsem si jistá jestli ho můžu použít, ale mě přišel dobrý. <https://www.yourgenome.org/facts/how-do-you-put-a-genome-back-together>
Image credit: Genome Research Limited - obrázek lze použít, nutná správná citace, popř. najít originál

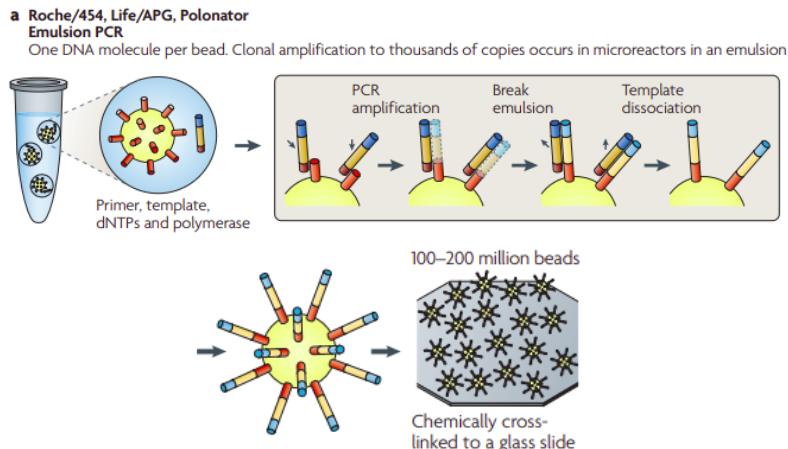


Obrázek 3.2: Single-end a paired-end read.

3.2.2 454 sekvenování a Ion Torrent

Pomocí 454 sekvenování je možné analyzovat více než milion molekul DNA najednou a délka každé jednotlivé sekvence se pohybuje okolo 700 až 1000 bází. V prvním kroku sekvenování je fragment DNA přichycena na malou "kuličku" na jejímž povrchu se postupně namnoží až kuličku zcela pokryjí identické fragmenty DNA. Následuje vložení kuličky i s DNA do jedné z

milionů komůrek na destičce s reakční směsí. Postup znázorněn na obrázku 3.3. V určitém momentě je do této směsy přidán vždy jen jeden typ báze. Mezi jednotlivými fázemi přidávání určité báze jsou přebytečné nukleotidy z předešlého kroku odstraněny. To znamená že v reakční směsí je vždy jen jeden typ nukleotidů. Během vložení každé nové báze do rostoucího řetězce DNA je uvolněna molekula zvaná pyrofosfát, která spustí několik chemických reakcí. V poslední fází enzym luciferáza vydá světelný záblesk, který je možné zachytit citlivou kamerou. Tento postup se nazývá pyrosekvenování. V případě, kdy je do řetězce přidáno několik stejných bází za sebou, například gen obsahuje podřetězec AAA, je vyzářeno, v našem případě, třikrát více světla než v případě jedné přiřazené báze. Kamera snímá celou destičku a na základě, která komůrka se rozsvítí pozná, kde proběhlo přidání báze. Intenzita světla pak určuje kolik bází bylo přidáno na jednou.



Obrázek 3.3: 454 sekvenování. [22]

Sekvenování Ion Torrent funguje na podobném principu sekvenování s rozdílem, že místo světla se měří změna pH v reakční směsí. Podle intenzity změny pH lze pak poznat kolik nukleotidů bylo přidáno do rostoucího řetězce.

Hlavní slabinou těchto dvou metod je značná chybovost při přidání mnoha stejných nukleotidů do řetězce za sebou. Například při přidání 10 A, nebude odpověď jednoznačná zda je to 10 A nebo 9.

3.2.3 Illumina

Při sekvenování pomocí Illumina jsou páry dvoušrobovice rozděleny na dva řetězce. Jednotlivé řetězce jsou následně přichyceny na malou destičku po-

mocí adaptéru. Každý řetězec se následně opakovaně množí až na destičce vznikne několik shluků. Přidání jedné molekuly ke druhé probíhá obdobně jako u Sanger sekvenování. Každý shluk tvoří jednu skupinu vzájemně identických řetězců. Mezi volné nukleotidy jsou opět zahrnutý nukleotidy označeny fluorescenční barvou za které nelze nic navázat. Oproti sangerovu sekvenování je ale tato blokace vratná a po přečtení citlivou kamerou dojde k odstranění blokující části molekuly. Počítáč si pak následně zpětně spočítá co to bylo za barvu (nukleotid). [5] [17]

3.2.4 SOLiD

SOLiD (Sequencing by Oligonucleotide Ligation and Detection) se spoléhá na enzym ligáza. Enzym je bílkovina, která určuje rychlosť chemických reakcí. Enzym ligáza konkrétně umožňuje připojení jednořetězcových molekul k stávajícím řetězcům. K teplátu jsou přidávány takzvané sondy, což jsou kousky DNA. Sondy začínají všemi možnými dvojkombinacemi čtyř základních nukleotidů. V součtu je 16 sond. Na každé sondě je jedna ze čtyř fluorescenčních barev. V jednotlivých krocích jsou sondy připojeny k rostoucímu řetězci. Následně je přečtena fluorescenční barva, která je odstraněna a může se tak navázat další sonda. Z výsledného signálu lze pak odvodit sekvenci DNA.

3.3 Metody třetí generace

Velkým rozdílem oproti druhé generaci je že DNA templát není před sekvenováním namnožen a je čten pouze z jedné původní molekuly. Existuje například PacBio od Pacific Bioscience, který k detekci využívá fluorescenčně značené nukleotidy. Díky jeho vysoké citlivosti je možné v reálném čase zachytit přidání i jediného nukleotidu do jediného řetězce DNA. Další zástupce je Oxford Nanopore jehož výhodou je jeho velikost. Oxford využívá odlišného tvaru bází. Obě metody jsou schopné přečíst přes 10 tisíc bází v rámci jedné analyzované molekuly DNA.

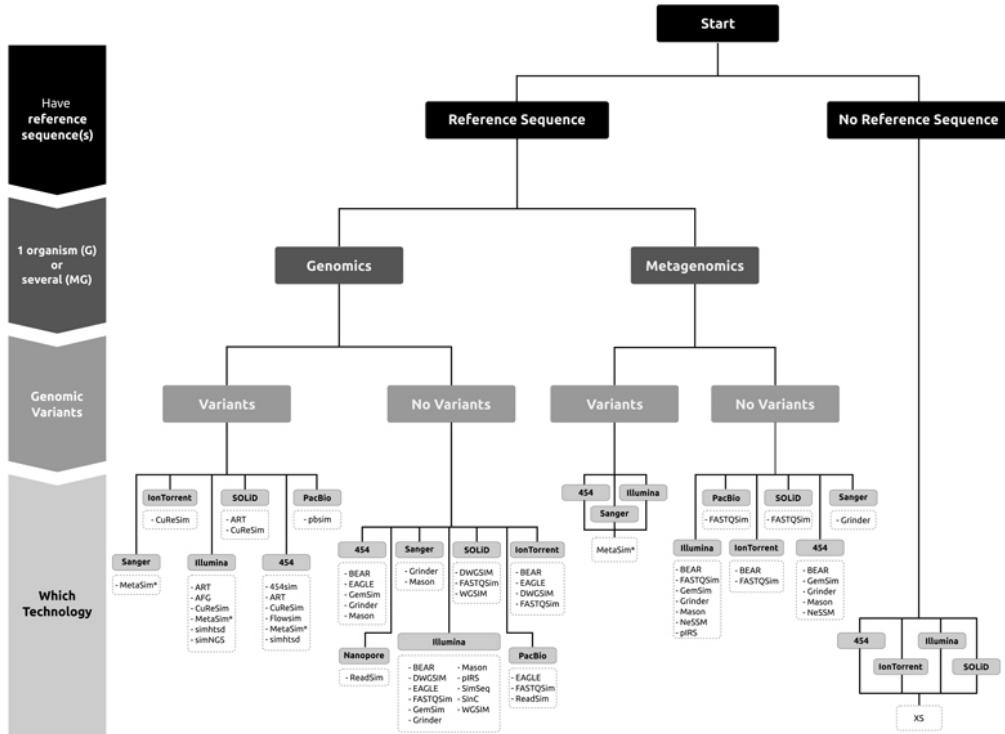
4 Analyza dostupných bioinformatických nástrojů pro zpracování NGS dat

NGS metody snižují náklady a zrychlují proces sekvenování za cenu kratších readů a menší přenosti což vedlo k mnoha bioinformatickým výzvám, jako je vytvoření nástrojů pro analýzu readů. Nástroje je možné mezi sebou porovnat pomocí realních nebo simulovaných dat. Přestože je validace na realních datech nezbytná, skutečné hodnoty na kterých jsou data založena jsou obvykle neznámá, což komplikuje jejich použití pro posouzení přesnosti (tj. jak blízko je odhadovaná hodnota ke skutečné hodnotě). Díky tomu je simulování dat čím dál více populární pro hodnocení, validaci či nastavování optimálních parametrů nástroje. [11]

4.1 Simulační nástroje pro generování syntetických readů

Dále uvedené nástroje byly vybrány na základě parametrů: simulování DNA, udržovány a volně dostupné. Následující informace vychází z článku [11] pokud nebude uvedeno jinak.

Většina simulátorů NGS vyžaduje referenční sekvenci ze které budou generovat ready. Tato referenční sekvence může být konkrétní genomická oblast, více zřetězených genomických oblastí, chromozom či celý genom. Některé simulátory vytváří zarovnání readů přímo do referenčního souboru (soubory SAM/BAM). Při používání simulátorů může být pro uživatele obtížné se rozhodnout kterou konkrétní hodnotu pro daný parametr určit nebo který vlastní profil vytvořit proto některé simulátory poskytují výchozí profily. Jedním z nich může být i generování chyb či model kvality. Nástroje jako jsou ART nebo SInC generují tyto profily na základě extrahovaných modelech ze skutečných dat. Nejčastější chyby jsou substituční a vložení či smazání (INDEL - insert-deletion).



Obrázek 4.1: Strom pro usnadnění výběru generátoru syntetických readů [11]

Data získaná z FN Plzeň/BC LF UK Plzeň byla sekvenována nástrojem Illumina proto je podle [11] na výběr z ART, AFG, CuReSim, Flowsim, MetaSim, simhtsd a simNGS. U simulátoru AFG je třeba chybové profily definovat ručně, CuReSim a MetaSim nejsou open source, simhtsd podporuje jen operační systém Linux, simNGS podporuje jen operační systémy Linux a MacOS. Proto byl vybrán simulátor ART který podporuje operační systémy Linux, Windows a MacOS. Dále generuje podle chybových profilů a profilů kvality které byly vytvořeny pomocí extrakce chyb získaných ze skutečných dat.

4.1.1 ART

ART (next-generation sequencing read simulator) je sada simulačních nástrojů, které generují syntetické ready, jako kdyby byly získány sekvenováním pomocí NGS. Nástroj ART dokáže simulovat single-end a paired-end ready ze sekvenátorů Illuminas, 454 společnosti Roch a SOLid od společnosti Applied biosystém. Ready, vytvořené nástrojem ART jsou používány pro testování a analýzů nástrojů zpracovávající právě NGS sekvence jako například zarovávání (nástroj Bowtie). Při použití nástroje ART je vstupním

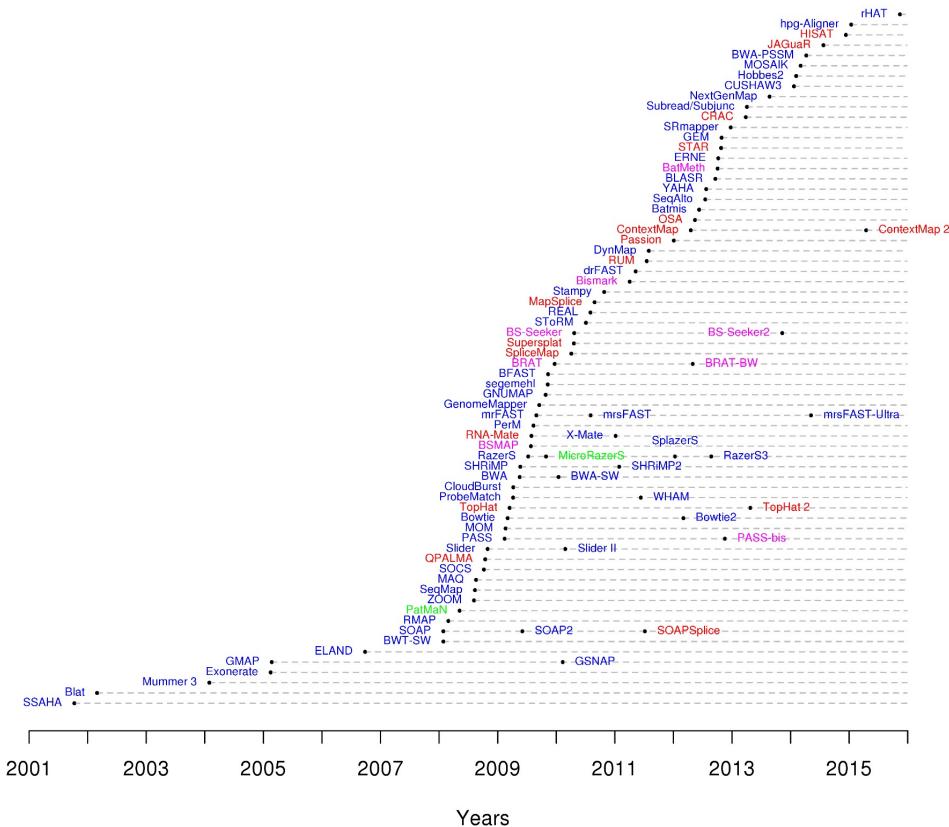
souborem sekvence genů na základě kterých jsou vygenerovány ready. [15]

Illumina je sekvenování založené na vratném umístění báze označené barvou do rostoucího řetězce jehož nejčastější chybou je substituce. Pravděpodobnost chyby substituce je určena na základě kvality skóre dané báze, které je závislé na pozici v rostoucím řetězci. Průměrné kvality skóre klesá v závislosti na zvyšování pozice báze. ART simuluje substituční chybu na základě tohoto skóre a empirického modelu získaného z trénovacích datasetů. INDEL chyba je simulována jen na základě empirického rozdělení z trénovacích dat a u Illuminy se vyskytuje jen zřídka. Pro paired-end simulaci, ART využívá dvou rozdílných kvality skóre pro každý pár readu jiný.

ART je implementován v jazyce C++ a je dostupný s licencí GPL verze 3 pro operační systémy Linux, MacOs a Windows. Je možné ho použít i jako C++ package. Pro jeho spuštění je nutný mít nainstalovaný kompilátor GNU g++ 4.0 nebo vyšší a knihovnu GNU gsl. Výstupy se čtou ve formátu FASTQ a zarovnání ve formátu ALN. může generovat zarovnávání také ve formátu SAM nebo UCS BED. Paired end ready jsou označeny stejným názvem souboru s 1 či 2 na jeho konci.

4.2 Nástroje pro zarovnávání readů

Zarovnávání bývá prvním krokem v mnoha genomických pipelinách. Často je to jejich nejpomalejší část, protože pro každý read musí zarovnávač vyřešit obtížný výpočetní problém. Určit pravděpodobné umístění v referenčním genomu. V současnosti je na výběr více než 90 nástrojů pro zarovnávání NGS readů. Nástroje jsou mezi sebou obvykle porovnávány na základě přesnosti a rychlosti mapování. V článku [7] bylo porovnáno 5 nástrojů pro zarovnávání DNA readů. Nástroj STAR měl narození od ostatních nástrojů menší přesnost, nástroj segemehl byl zase náročný na paměť (podle článku až 70 GB) což se na stolním počítači těžko dosahuje. Ze zbývajících nástrojů byl vybrán nástroj Bowtie2 díky jeho rychlosti vzhledem k nástroji BWA, která vzhledem k množství experimentů bude přínosem.



Obrázek 4.2: Nástroje pro zarovnávání NGS readů. [7]

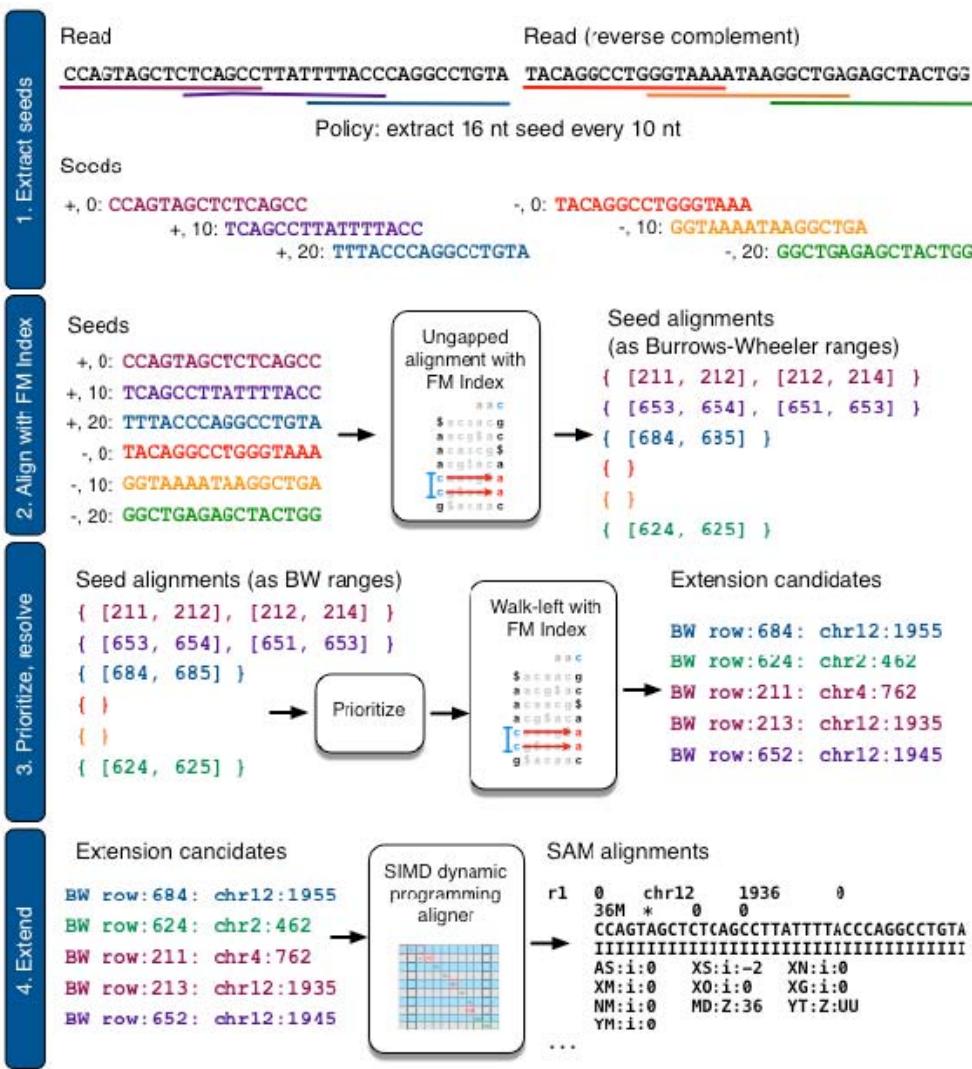
4.2.1 Bowtie2

Bowtie2 je rychlý a paměťové efektivní nástroj pro zarovnávání krátkých sekvencí DNA na velké genomy. Bowtie2 je schopný zarovnat více než 25 milionů readů dlouhých 35 bp za hodinu (při běhu na jednom CPU) pro lidský genom s malým využím paměti. Bowtie2 využívá FM indexaci s Burrows-Wheeler transformací (BWT) a přidává k ní backtracking pro sledování nekonzistence. Novější verze Bowtie2 by měla být oproti Bowtie1 citlivější a rychlejší na delší ready než je 50 nukleotidů a navíc je oproti první verzi schopná se vypořádat z chybami vložení či smazání báze způsobené sekvenováním. Na lidský genom potřebuje Bowtie2 3.2 gigabajtů RAM. Nástroj bowtie je implementovaný v jazyce C++ s použitím knihovny SeqAn a je open source. Podporuje standardní vstupní formáty FASTQ a FASTA. Výstupní zárovnání z Bowtie je ve formátu SAM, což umožňuje návaznost s dalšími nástroji jako je třeba SAMtools. Následující informace vychází z článků [20] a [19] pokud není uvedeno jinak.

Mnoho zarovnávačů používá indexy k rychlému snižování kandidátů pro umístění zarovnáváního readu. Bowtie vytváří indexy referenčních genů permanentní a lze je tak použít napříč běhy. Algoritmus FM indexu obvykle funguje na vyhledávání přesné shody. V případě hledání umístění readů na referenční gen není toto řešení použitelné, protože ready mohou obsahovat chyby vzniklé sekvenováním případně genové mutace. Proto Bowtie každé zarovnání zakládá na kvalitě znaku báze v daném readu. Bowtie postupně vytváří dlouhý sufix. Pokud se sufix nevyskytuje v textu pak se může algoritmus vrátit a v již vytvořeném sufixu nahradit bázi zajinou. Dále pokračuje obdobným způsobem. Tento způsob změny báze je dále označován jako backtracking. Pokud by měl algoritmus na výběr substituovat za více bází vybere tu s nejnižší kvalitou znaku v readu. Protože Bowtie algoritmus v základu bere první přijetelné řešení je možné, že jeho nalezené řešení není to nejlepší. Pro nalezení toho nejlepšího řešení je třeba použít přepínač $- - best$, jeho funkčnost je ale na úkor rychlosti, která může být 2x či 3x pomalejší. Bohužel nemůže být tento přepínač použit u paired-end readů. Zároveň je možné nastavit maximální počet nahrazených bází v readu. [20]

V případě že backtracking mechanismus není úspěšný může docházet k jeho nadměrnému vyskytu. Bowtie2 se tento jev snaží zmírnit dvojím indexováním. První index obsahuje BWT genomu a je označován jako dopředný index. Druhý obsahuje opět BWT genomu, ale se znaky v sekvenci v opačném pořadí, označovaný jako zrcadlový index. Read je pak v půlce rozdělen na dvě části a jejich zarovnávání probíhá odděleně tak, že je vždy backtracking povolen jen v dané části, která je zrovna zarovnávána. Pravá část je zarovnáváná podle dopředného indexu a levá část je zarovnávána podle zrcadlového indexu.

Předchozí algoritmus funguje dobře pouze v případě, kdy reference nebo read neobsahují mezery (báze chybí nebo naopak přebývá). Proto byl algoritmus rozšířen, jak je popsáno následovně.



Obrázek 4.3: Algoritmus zarovnání readů s mezerami. [19]

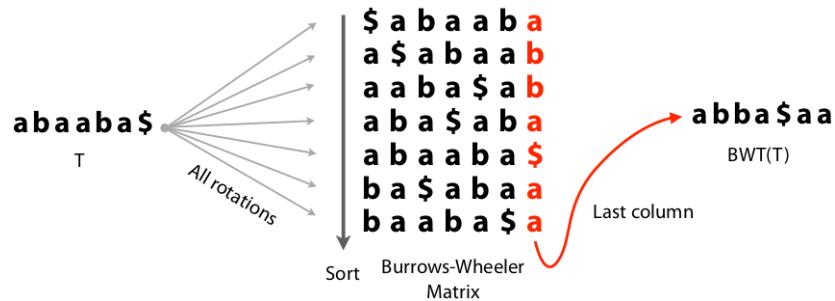
Pro každý read

1. Extrahování seedu (podřetězce readů) z readu a jeho převrácených doplňku
2. Seedy jsou zarovnány na refenci v bezmezerové modelu za pomocí full-text minute indexu. Čísla v závorkách značí rozsah řádků v Burrows Wheeler matici kam byl seed zarovnán
3. Zarovnání seedů je seřazeno offsetu tu na refrenčním genomu.
4. Ready jsou zarovnány. Díky předchozím krokům je značně omezen prostor kam mohou být zarovnány. Pro zvýšení výkonu je použito SIMD (accelerated dynamic programming).

4.2.2 Burrows-Wheeler transformace

Burrows-Wheelerova transformace (BWT) je reverzibilní permutace řetězců v textu. Původně byla používána pro kompresy dat. Indexace založená na BWT umožňuje efektivní vyhledávání ve velké textu s malou paměťovou náročností.

BW transformace řetězce T , $BWT(T)$, je zobrazena na obrázku 4.4. Znak $\$$ je připojen na konec řetězce a zároveň musí platit, že se tento znak se v řetězci nevyskytuje. Burrows-Wheeler matice řetězce T je konstruovaná jako všechny cyklické rotace řetězce T , které byly seřazeny podle abecedy, kde znak $\$$ se bere, že je na začátku abecedy. Výstup, $BWT(T)$ pak představuje poslední sloupec matice. Tento řetězec má stejnou délku jako původní řetězec T . [20]



Obrázek 4.4: Burrows-Wheeler transformace řetězce T . [18]

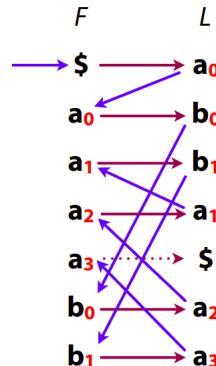
Burrows-Wheeler matice má vlastnost, která se nazývá last first mapping (LF). To znamená, že i-tý výskyt znaku X v prvním sloupci je i-tý výskyt znaku X v posledním sloupci. V případě přidání indexu do řetězce T je toto pravidlo pro znak a zobrazeno na obrázku 4.5. Obdobně to platí i pro ostatní znaky v řetězci.

$$T = a_0 \ b_0 \ a_1 \ a_2 \ b_1 \ a_3 \ \$ \quad (4.2.1)$$

<i>F</i>	<i>L</i>
\$ a ₀ b ₀ a ₁ a ₂ b ₁ a ₃	
a ₃ \$ a ₀ b ₀ a ₁ a ₂ b ₁	
a ₁ a ₂ b ₁ a ₃ \$ a ₀ b ₀	
a ₂ b ₁ a ₃ \$ a ₀ b ₀ a ₁	
a ₀ b ₀ a ₁ a ₂ b ₁ a ₃ \$	
b ₁ a ₃ \$ a ₀ b ₀ a ₁ a ₂	
b ₀ a ₁ a ₂ b ₁ a ₃ \$ a ₀	

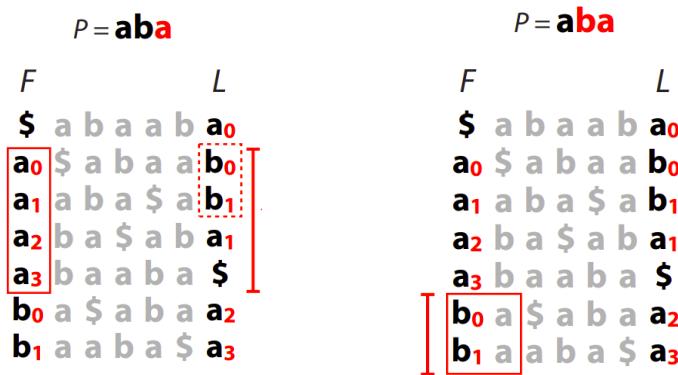
Obrázek 4.5: Burrows-Wheeler transformace last first mapping (LF). [18]

Zpětné získání řetězce je znázorněno na obrázku 4.6. L sloupec je řetězec, který je výstupem BW transformace. F sloupec je snadné na základě L sloupce odvodit. Jelikož platí pravidlo, že počet jednotlivých znaků je stejný, stačí je pouze přemístit do F sloupce a seřadit podle abecedy. Dále s využitím LF je řetězec získán zpět. Jako první se vezme přidaný znak \$. Ve stejném řádku ve sloupci L se nachází a_0 . To znamená že řetězec začíná \$.a. Algoritmus pokračuje s a_0 v F sloupci. Ve stejném řádku v L sloupci je b_0 . b_0 je přidáno do řetězce a pokračuje až do doby než by byl opět znak \$.



Obrázek 4.6: Burrows-Wheeler transformace zpětné získání původního řetězce. [18]

Díky vztahu mezi F a L sloupcem je možné vyhledávat daný řetězec (zobrazeno na obrázku 4.7). Například vyhledávány řetězec bude $P = aba$. Při pohledu do F sloupce jsou nalezeny všechny sloupce začínající a , následně v L sloupci ve stejných řádcích jsou nalezeny dva výskytu b . Již je získán sufix ba , který existuje. Pokračuje se dále na řádky, které začínají právě nalezenými b . V sloupci L pro dané řádky jsou nalezna a . Řetězec $P = aba$ se v textu vyskytuje.



Obrázek 4.7: FM index - získání prefixu. [18]

4.3 Další pomocné metody

4.3.1 Levenshteinova vzdálenost

Levenshteinova vzdálenost zjišťuje rozdílnost dvou textů na základě počtu změn, které je třeba udělat, aby bylo z jednoho řetězce získán druhý řetězec. Za úpravy se považuje vložení, smazání a nahrazení. Algortimus funguje tak, že se snaží ze slova, které bylo jako první v argumentu vytvořit slovo předáno jako druhé. Příkladem může být vzdálenost mezi řetězci *SPAM* a *PARK*. Vzdálenost těchto slov je 3. Výstup v případě python knihovny je možné vidět následovně. Výstup 4.3.1 je v případě *SPAM*, *PARK*. Výstup 4.3.2 je v případě *PARK*, *SPAM*. Změny jsou definovány: o jakou změnu jde, index znaku v prvním řetězci a index znaku v druhém řetězci. Je možné si všimnout závislosti mezi těmito dvěma postupy. [32]

$$('delete', 0, 0), ('insert', 3, 2), ('replace', 3, 3) \quad (4.3.1)$$

$$SPAM -> _PAM -> _PARM -> PARK$$

$$('insert', 0, 0), ('delete', 2, 3), ('replace', 3, 3) \quad (4.3.2)$$

$$PARK -> SPARK -> SPAR_- -> SPAM$$

5 Implementace

5.1 Popis problému

máme krátkou délku že read který dostáváme jsou 250 bp dlouhé a jeden gen může být dlouhý 14738 bp, akorát že z nemocnice ti dají 251 s tím že jednotlivé ready se nám tedy mohou překrývat- tohle si nejsem jistá jestli se můžou překrývat můžou tam být chyby

pak by se tam dala přidat heurestika že bych brala známe haplotypy

Možná pak ještě pracovat s pravděpodobností výskytu daného genu

jenže bowtie může klidně někam zarovnat tam kam to ve skutečnosti napatří protože tam hledá třeba backtracking a nebo vložení a smazaní chybu

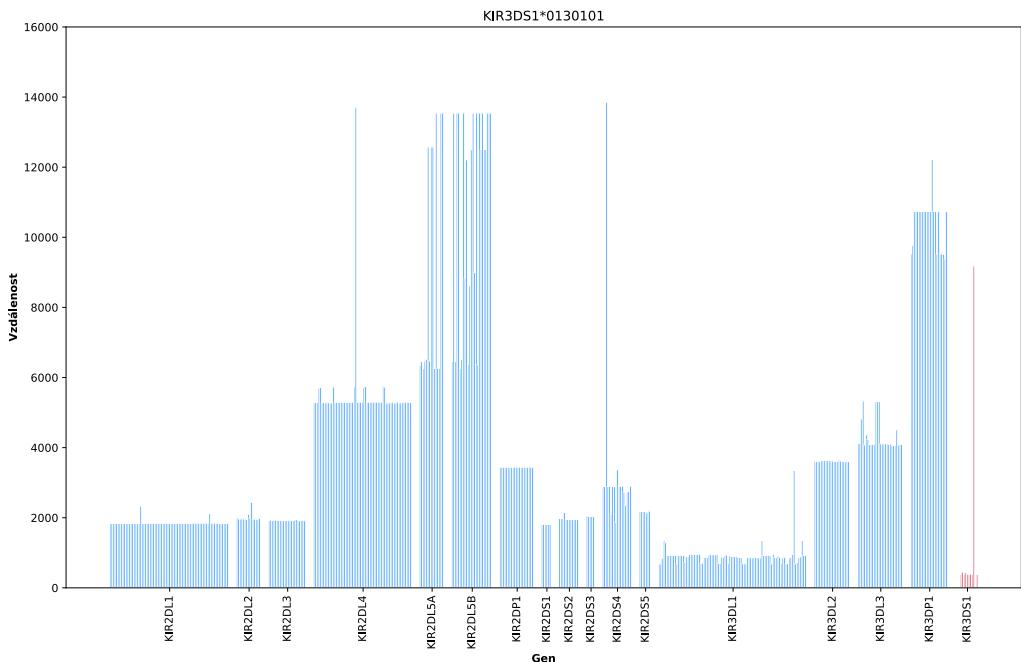
asi sem dopsat že i ty alely pro jeden gen můžou být různě dlouhé protože tam probíhají mutace Úkol: vyhodnocovací algoritmus NGS dat pro identifikaci alel na úrovni relevantního coding-regionu, ale porovnání i na non-coding

5.2 Referenční geny

Referenční geny byly převzaty z IPD-KIR [28] konkrétně soubory ve formátu *fasta* uloženy ve stejnojmenné složce. Jednotlivé soubory jsou pojmenovány genem, který obsahují např. *KIR2DL1_gen.fasta*. Každý soubor představuje všechny dostupné alely konkrétního genu. Jedinou výjimku tvoří soubory *KIR_gen.**, které obsahují všechny geny a navíc i pseudogeny.

Kromě souborů **_gen.fasta* obsahuje složka *fasta* také soubory **_prot.fast* a **_nuc.fasta*. Soubor *_gen.fasta* obsahuje informace o celých genech. Oproti tomu *_nuc.fasta* obsahuje nucleotidy, tedy pouze exony bez intronů. Soubor **_prot.fast* obsahuje sekvence proteinů, které vznikly z RNA. Data získaná z nemocnice budou odpovídat alelám uvedených v *_gen.fasta*.

Při analýze porovnávání souboru *nuc* a *gen* bylo zjištěno, že v souboru *nuc* je více alel než v souboru *gen*. Konkrétně v souboru *gen* je 461 alel a v souboru *nuc* je 1109 alel. Nejmenší Levenshteinova vzdálenost mezi alelami je 1, největší 15 943 a průměrná 4768.98. V práci bude jako reference použit soubor *_gen*.



5.3 Testovací genotypy

Genotypy, na kterých byl nástroj testován byly dodány vedoucí práce. Genotypy test1 - test11 byly sestaveny podle definovaných genotypů a umýslně vybrány ty které představují nějaký problém.

TODO co je sakra: KIR3DS1*049N ???? - Kde jsem to našla v nuc nebo gen?

TODO: KIR_gen.fasta includes the DNA sequence for all alleles, which have genomic sequences available.

TODO sem asi dodat ty věci ohledně dat z nemocnice

TODO ještě dodat že tady asi nebudou všechny známí alely. Musíme porovnat nuc a gen jak jsou ty geny s alelama mezi sebou

5.4 Reálná data

Pro validaci algoritmu na reálných datech kompletní KIR genom u 9-ti komerčních linií byl ve spolupráci FN Plzeň – LFP UK amplifikován z izolované DNA pomocí long-range PCR za použití směsi 6 primerů. Sekvenační knihovna je připravena modifikovaným protokolem NEBNext[®] Ultra TM II

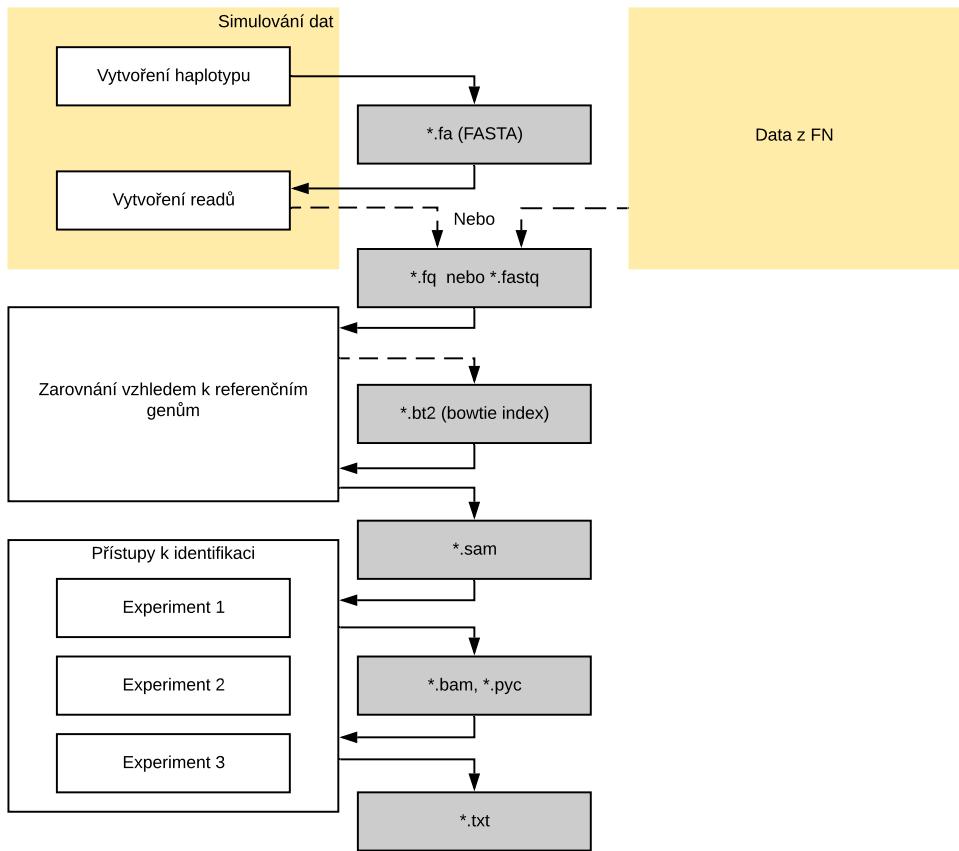
FS DNA Library Prep Kit for Illumina (New England Biolabs) a knihovny jsou sekvencovány v pair-end režimu na přístroji Illumina Miseq s pokrytím 100. [21]

5.5 Návrh systému

Systém byl navržen jako modulární, díky tomu je možná jednoduchá náhrada jakékoliv jeho části.

Vše začíná získáním dat pro která má být vyhodnoceno, které KIR alely obsahuje. Buděj možné dostat přímo data z Fakулní nemocnice či biomedicínského centra. To jsou data na kterých bude prováděna verifikace nástroje. Druhou možností je data vyrobit. Na těchto datech byl nástroj vyvíjen a laděn. Data mohou být vyrobena ručně nebo je lze vyrobit za pomoci programu. V dalším kroku musí být haplotyp "rozbit" do podoby jako by vyšel ze sekvenátoru. Rozdělí se na ready a vytvoří s v něm chyby. To se provádí za pomoci nástroje ART.

V následující části jsou získaná data, tedy ready, zarovnána na referenční genom pomocí nástroje Bowtie. Nakonec je zarovnání vyhodnoceno a rozpoznáno o jaké alely genů se pravděpodobně jedná. Vyhodnocení je rozděleno do několika experimentů. Pro zjednodušení práce s výsledky je doplněn krok, kdy jsou názvy alel podle pořadových čísel nahrazen na názvy alel podle jejich skladby.



Obrázek 5.1: Návrh systému. TODO to vyhodnocení chce předělat. Chybí Bowtie indexy

5.5.1 Použité programové prostředky

Python

Program byl navržen a implementován na operačním systému Linux za použití především programovacího jazyku Python. Pro spuštění programu je nutné mít nainstalovaný Python ve verzi 3.8. TODO dodat knihovny

5.5.2 Bordel

Co znamená konec 3 a konec 5?

5.6 Modulové jednotky programu

Vše potřebné pro samotný běh programu obstarává skript *run.py* spolu s nastavením v souboru *config.py*. Skript *tun.py* postupně pouští jednotlivé moduly. Díky nastavení v *config.py* je možné si zvolit spuštění jen některých modulů. Například pouhé vytvoření testovacích dat nebo pouze jejich zarovnání a v neposlední řádě pouštět vyhodnocování zarovnávaných dat.

5.6.1 Config

S konfiguračním souborem *config.py* jsou spojeny všechny skripty a obsahuje jejich veškerá nastavení. Jak již bylo zmíněno je možné pomocí tohoto nastavení spustit jednotlivé moduly. Jedná se především o položky *CREATE_READS*, která udržuje informaci o spuštění vytvoření syntetických readů. *ALIGN* starající se o spuštění zarovnání a *EVALUATE*, která řídí spuštění vyhodnocení zarovnání. Důležitou položkou v configu jsou cesty ke zdrojovým a výstupním složkám. Dalším nastavením je obsah genomů při případném vygenerování testovacích dat.

5.6.2 Simulování dat

O simulování dat se stará skript *create_syntetic_reads.py* a je rozděleno na dva kroky: vytvoření genomů a vytvoření readů. Mezi těmito dvěma fázemi vzniká soubory s příponou *.fa*. Každý tento soubor obsahuje právě jeden KIR genom. Tyto genome jsou následně použity jako vstupní soubor pro vytvoření readů, které se provádí za pomocí nástroje ART. Pro vytvoření genomů je volán skript *create_genome.py*, který vytvoří genome na základě nastavení v configu pod položkou *GENOMES* a za pomoci referenčních genů v souboru pod položkou v configu *REFERENCE_KIR_GENS_FILE*. Vytvoření probíhá obdobně jako je popsáno níže u ručního vytvoření testovacího genomu. Výsledné genome jsou uloženy do adresáře z configu *GENOME_FOLDER*. Výstupem modulu *create_syntetic_reads* je soubor s příponou *.fq*, který by měl odpovídat formátu reálných dat které byly dodány. Výstupní soubory jsou uloženy do složky pod proměnnou *READS_FOLDER*.

Ruční vytvoření testovacího genomu lze udělat následujícím způsobem. V prvním kroku je v referenčním souboru vybrána konkrétní alela genu. Někdy je možné najít shodu kdy se alely liší jen v konečné fazí jejich označení a v genomu je pouze první 5 čísel. V tomto případě může být vložena jakákoli z těchto alel. Vkladaná alela musí být vložena včetně její hlavičky tedy: *>KIR:KIR00138 KIR3DL3*0040201 12390 bp.*

5.6.3 Zarovnání vzhledem k referenčním genům

Zarovnávání obstarává skript *alignment_reads_to_reference.py* s pomocí nástroje Bowtie2. V nastavení je nutné vyplnit *BOWTIE_HOME_DIRECTORY* podle umístění nástroje Bowtie na konkrétním počítači. V prvním kroku jsou vytvořeny Bowtie indexy, které je možné použít na příč běhy, proto je v nastavení položka *BOWTIE_BUILD_INDEX*, díky které je možné toto vytvoření povolit nebo zakázat. Bez vytvořených indexů, tedy indexů z minulých běhů, ale Bowtie nebude zarovnávat. Bowtie vytváří indexy na základě obsahu souboru *REFERENCE_KIR_GENS_FILE*. V dalším kroku jsou načteny všechny ready ze složky uvedené v *READS_FOLDER* a následně je na ně puštěn nástroj Bowtie. Tady je nutné aby byli ready paired end a to tedy aby se vyskytovali dvakrát jednou z 1 na konci a podruhé s 2. V základu tento předpoklad zajistí správné nastavení ARTU, který je takto nastaven. Výstupní soubory jsou ve formátu *.sam* a jsou umístěny dle položky *ALIGMENT_FOLDER* v nastavení.

5.6.4 Přístupy k identifikaci alel

Detailní výsledky pro všechny genotypy je možné najít v příloze. V následující teoretické části jsou uvedeny vždy simulované genomy.

Experiment 1

Jako první pokus o určení alel, které by mohly být obsaženy v genomu bylo pouhé oříznutí alel, které by měly větší procentuální šířku zarovnání (například 90%). V případě buněčné linie AMALA, která byla simulována nástrojem ART a která obsahuje 19 alel je výsledek zobrazen níže. Červeně jsou označený alely, které se v genomu skutečně nacházejí, číslo v závorce udává jejich procentuální pokrytí.

- 3DL2*0070102 (99.64%)
- 2DS4*0010101 (99.64%)
- 2DS4*0010102 (99.55%)
- 2DS5*0020101 (99.50%)
- 3DL2*0020105 (99.48%)
- 2DS2*0010101 (99.43%)
- 3DL3*00802 (99.34%)
- 3DL2*0070103 (99.29%)
- 3DL3*0040201 (99.22%)
- 3DL2*0020106 (99.15%)
- 3DP1*007 (99.04%)
- 2DS4*0010107 (98.99%)
- 2DS5*0020103 (98.98%)
- 3DS1*055 (98.70%)
- 3DL2*0020102 (98.40%)
- 3DL2*0020104 (98.19%)

- 3DL2*018 (98.07%)
- 2DL2*0030101 (98.07%)
- 2DP1*0020105 (97.99%)

Následovně jsou uvedeny alely, které do genomu patří, mají pokrytí více než 90%, ale nejsou v prvních 19.

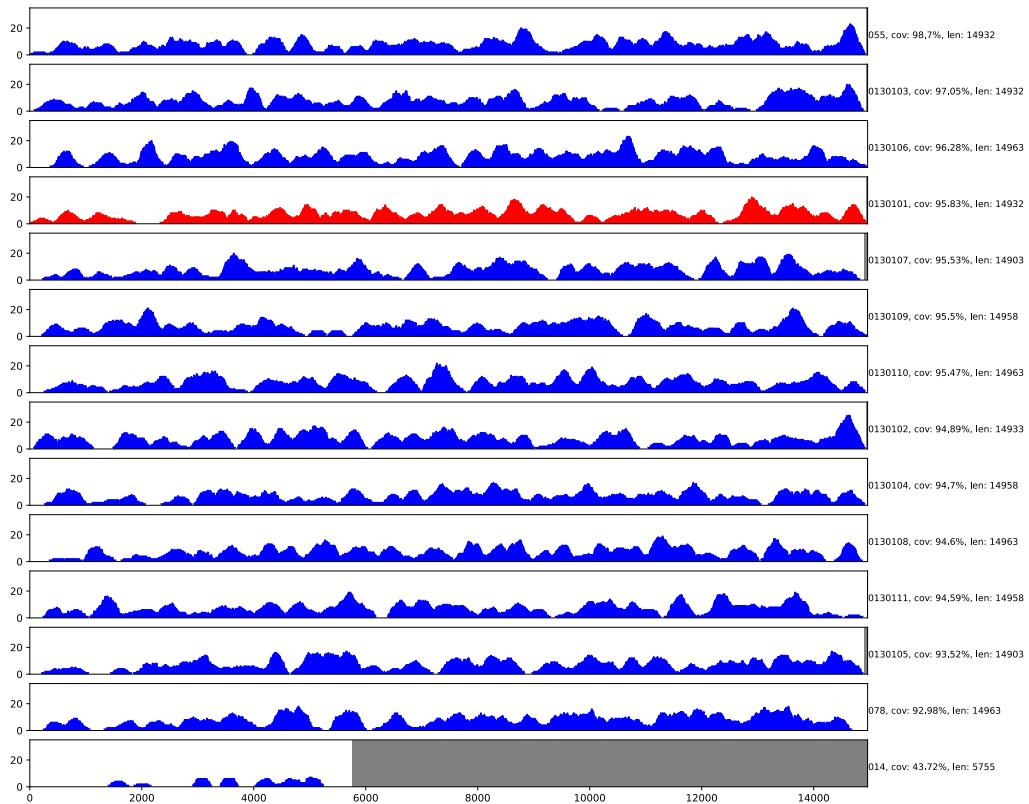
- 2DL2*0030102 (97.29%)
- 2DL3*0010109 (94.16%)
- 2DL5A*00102 (96.6%)
- 2DS1*0020106 (96.57%)
- 3DP1*0090101 (96.46%)
- 3DS1*130101 (95.83%)

Nakonec jsou uvedené alely, které do genomu patří, ale mají pokrytí menší než 90%.

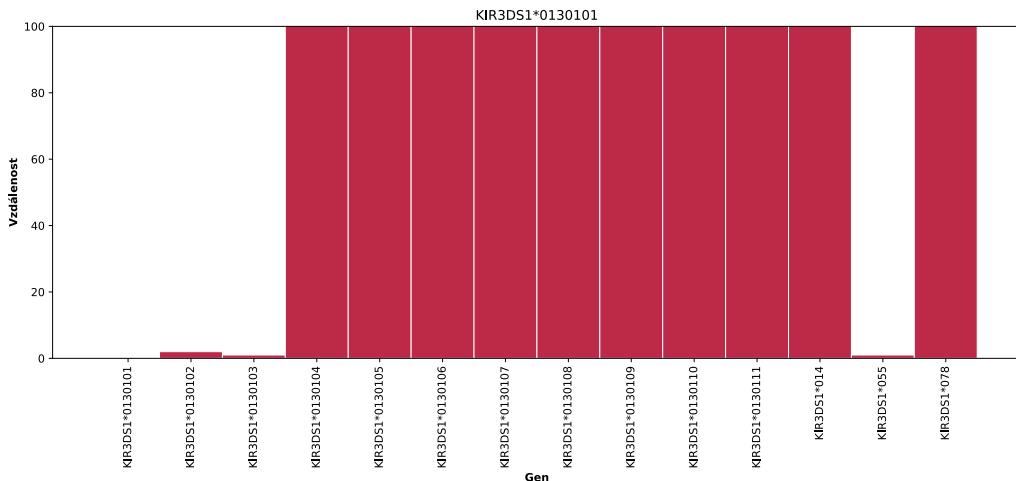
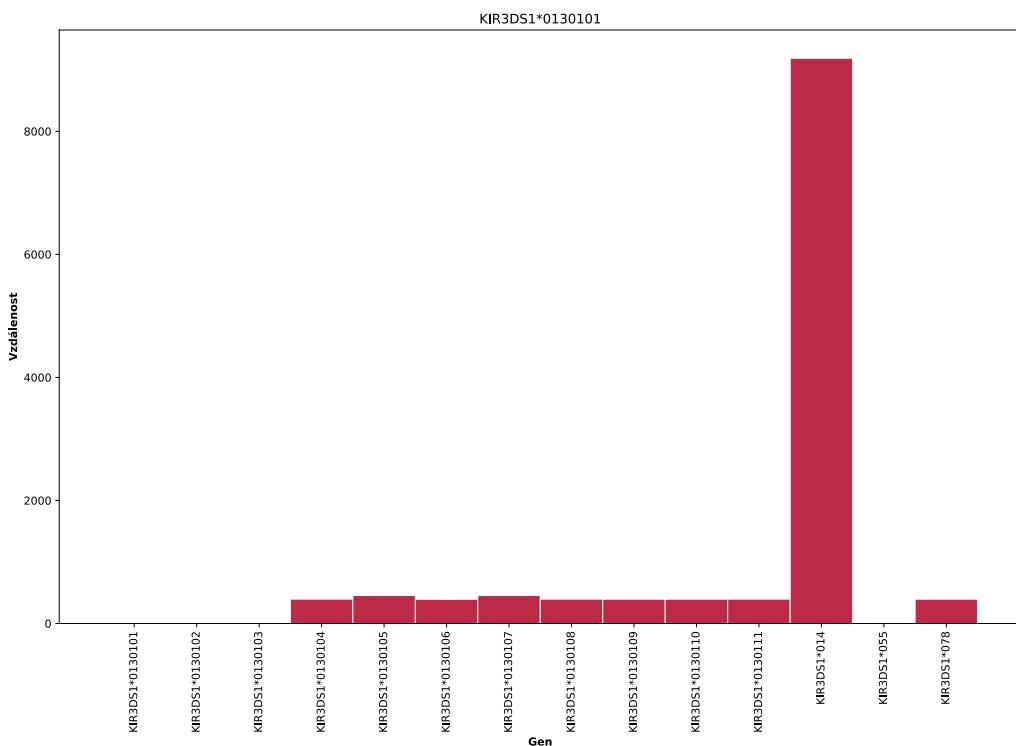
- 2DL1*0030201 (81.8%)
- 3DL1*0150201 (70.71%)

Ná následujícím obrázku jsou zobrazeny všechny alely genu KIR3DS1 a jejich zarovnání. Červeně je označena alela, která do genomu amala patří. Je možné si povšimnout, že alela s největším pokrytím do genomu nepatří. Navíc pokrytí větší než 90% mají všechny alely až na jednu. Neméně důležitá část je, že poslední alela má značně kratší délku než všechny ostatní alely stejného genu. Poslední věcí je hloubka pokrytí, která díky nastavení sekvenátoru by měla odpovídat hladině 100 u alely, která do genu patří. Jak je vidět z obrázku hloubka pokrytí se u většiny alel pohybuje v extrémech kolem 20.

KIR3DS1



Všechny vyše uvedené analýzy navádí na bližší prozkoumání podobnosti alel. Níže jsou uvedeny grafy Levenshteinovy vzdálenosti ostatních alel stejného genu vzhledem k alelu 3DS1*0130101, tedy té která je obsažena v genomu.



Z předchozích obrázků je vidět, že 3 alely, jsou vzálené v rámci jednotek, 9 alel je vzdálených kolem 500 a jen jedna alela je více rozdílná, což pravděpodobně způsobuje to, že je kratší. Přichází tedy na řadu otázka jak určit, které alely jsou si až moc podobné a pak z nich rozhodnout kterou odstranit. Vzdálenost kdy jsou alely blízké je možné nastavit v configu. Dále je porovnáváno jejich pokrytí v místech kde jsou rozdílné. Pokud je hloubka pokratí jedné alely 2krát větší než hloubka pokrytí druhé alely je druhá alela odstrňena.

V níže uvedené tabulce jsou uvedeny výsledky prvního experimentu s parametry pokrytím větší než 90% a považovány za blízké v případě vzdálenosti menší než 100. Jedna se opět o genom AMALA který obsahuje celkem 19 alel a žádná z nich se neopakuje. Krok 1 je pouze statistika prvního zarovnání. Obsahuje všechny alely které jsou v referenci. Celkově je v referenci 461 alel. Pouze dva geny ze všech uvedených v referenci se nenacházejí v amale. Krok 2 je po odstranění alel které mají pokrytí menší než 90%. V posledním kroku 3 jsou uvedené výsledky po odstranění podobných.

Zbývá alel	Krok 1		Geny navíc	Zbývá alel	Krok 2		Geny navíc	Zbývá alel	Krok 3		Geny navíc			
	Ztraceno				Ztraceno				Ztraceno					
461	0	-	2	2DL5B 2DS3	113	2	2DL1*0030201 3DL1*0150201	0	-	23	4	3DP1*0090101 2DL4*0010201	0	-

Tabulka 5.1: Výsledky experimentu1 u genomu amala. Odřezány byly alely, které měli pokrytí menší než 90%. Alel u genotypu značí počet v daném genotypu. Číslo v závorkách udává kolik alel je dvakrát v daném genotypu. V každém kroku zbývá alel je z kolik alel ještě zůstalo ve výběru, ztraceno určuje kolik alel má být v genotypu, ale algoritmus je vyřadil. Za tímto číslem jsou vypsané alely které byly ztraceny. V dalších krocích jsou vypsány alely bez těch které už byly ztraceny v předchozích krocích. Obdobně je to s geny navíc, které udávají počet a jaké geny již neobsahují žádnou z alel, která nalezní do daného genomu.

TODO Tady možná zmínit, že tedy stačí zarovnávat na těch 5 z praktického hlediska je daleko hroší varianta, že to identifikuje gen, který tma není než že se nějaký ztratí

todo dodat ty průměry, nejhorší a nejlepší

Experiment2

Experiment2 byl navržen s iterační zarovnávání. Mělo by pak být v dalších krocích jasnější která alela do genomu skutečně patří a která ne. V první fázi se provede zarovnání na celou KIR referenci. V prvním kroku je vytvořena statistika obdobně jako v prvním experimentu a odstraněny alely, které mají menší pokrytí než *CUT_COVERAGE_ALELS* které je uvedené v *configu*. Ze zbylých alel je vytvořena nová reference na kterou je provedeno nové zarovnání readů. Opět je vytvořena statistika. Výsledky této statistiky nalezneme pod krokem 2. Následuje krok 3 v němž soutěží podobné alely v rámci jednoho genu. Podobnost alel se bere na základě Levenshteinové vzdálenosti. Alely jsou si podobné v případě, kdy je jejich vzdálenost menší

než vzdálenost uvedené v *configu* pod parametrem *CLOSE_DISTANCE*. Následuje vytvoření nové reference a nové zarovnání. odstraněna.

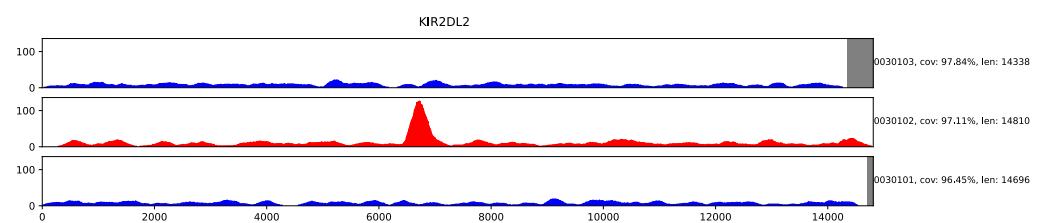
Na následujících obrázcích je vidět alela 2DL1*0030201 genomu AMALA. První obrázek značí její zarovnání po prvním kroku. Druhý obrázek značí její zarovnání po druhém kroku. V tomto případě bylo zarovnání jednoznačně ku prospěchu.



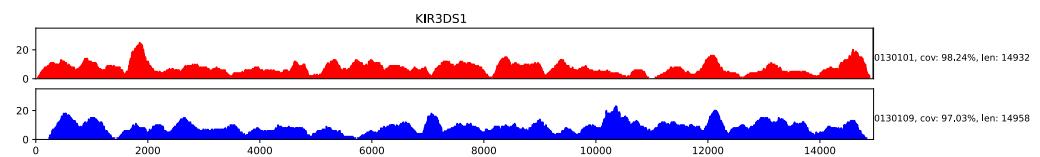
Ovšem je možné se potkat i s alelami, kde pokrytí zarovnání klesne, jako v případě 2DL3*0010109 genomu AMALA.



Na následujícím obrázku je výsledek genu 2DL2 po třetím kroku. Alela která do genomu patří má jasný vrchol. Tento vrchol označuje místo kde bylo zarovnáno mnoho readů.



Oproti tomu u genu 3DS1 po kroku 3 jsou zbývající alely k nerozeznání.



Proto byl vymyšlen způsob, krok 4, který odstraní alely v případě kdy sdílejí gen s alelami s jednoznačným vrcholem, ale u genů kde to není jasné nic

odstraňovat nebudou. Pro každý gen je spočítána hloubka pokrytí pro jednu alelu. Z jednoho genu jsou porovnávány každá alela s každou. Pokud jedna alela má 2krát větší maximální hloubku pokrytí a zároveň je maximální histogram této alely 2krát větší než průměr

Experiment 3

Další experiment byl založen na omezení stavového prostoru a to vytvořením clusterů. Alely jsou seřazeny od nejvíce pokryté po nejmenší. První nejvíce pokrytá je vložena do prvého clusteru. Další alely projdou všechny clustery a pokud je alespoň jedna alela vzdálenost s právě procházející menší než *CLUSTER_DISTANCE* uvedené v *configu* je alela přiřazena do tohoto clusteru. Pokud ne pokračuje v porovnávání dokud nezkontroluje všechny clustery. V případě, kde nezapadá ani do jednoho z clusterů, vytvoří nový.

Při vzdálenosti 5 bylo vytvořeno kolem 224 clusterů může jich být o pář jednootek více nebo méně z důvodu, že alely nejdou za sebou ve stejném pořadí, ale v tom pořadí od nejvíce pokrytých po nejméně pokryté. Při vzdálenosti 10 vznikalo kolem 171 clusterů, při vzdálenosti 15 vznikalo 143 a nakonec při vzdálenosti 30 vznikalo kolem 122 clusterů. Clusteterování ve smyslu jako je navržené zde, nepřineslo očekávaná zlepšení. Je třeba se nad clustrováním alel zamyslet jinak. Z příkladů největší clusterů je možné si povšimnout, že většinou alely pocházely ze stejných genů.

návrh na experiment 4

Udělat clustery chytřejší ... já potřebuju dát dohromady ty alely který si mezi sebou berou ready to znamená hledat kolik z nich má minimální rozdíl v 251 dlouhem useku oproti jine protože ngs a takovy jsou vstupni data a ted s jakyma se trefuje a pak kolik procent ty alely se tam da trefit problem vypocetni narocnost

5.6.5 Překlad alel

Při použití Bowtie vyvstal problém kdy jsou výsledné alely pojmenovány pořadovým číslem jak byli objeveny nikoli názvem vyjadřující jejich skladbu. Proto vznikl modul překlad alel z pořadových čísel do názvů vyjadřující skladbu alely obsahující skript *renaming_alels_result.py*, který projde všechny soubory ve složce pod proměnou *RESULT_FOLDER* a nahradí pořadová čísla příslušným názvem. Nakonec nahradí původní soubor, takto upraveným souborem.

6 vyhodnocení mezi výsledku a nastavení parametru

tohle asi pak dát spíš k vyhodnocení zarovnání Krok jedna je jen vyhodnocení prvního zarovnání to asi nemá moc smysl sem dávat Krok dva je jenom odstranění těch které mají menší zařazení než to uvedené v configu Krok tři je odstranění stejných a Krok 4 je odstranění průměrů ta tabulka je horizontálně velká a zasahujeme mi do číslování stránky a i mi přesahuje vpravo. do popisku připsat že krok 2 tak ty vypsaný alely jsou navíc k těm z kroku jedna

7 Porovnání přístupů k identifikaci a parametrů

Porovnání jednotlivých přístupů probíhalo pomocí přesnosti (precision) a úplnosti (recall). Díky přesnosti je možné odhadnout jak moc jsou výsledky relevantní. Naopak pomocí úplnosti je možné odhadnout kolik skutečně relevantních výsledků bylo přiřazeno. Při převyšující úplnosti bylo získáno mnoho alel, ale mnoho jich do genomu nepatří. Naopak při převyšující přesnosti nad úplností je většina vybraných alel v genomu ale mnoho jich bylo ztraceno. Proto je snaha balancovat přístup tak aby byly obě hodnoty dosahovaly co nevyjvýšších procent.

$$Prec = \frac{TP}{TP + FP} \quad (7.0.1)$$

$$Rec = \frac{TP}{TP + FN} \quad (7.0.2)$$

	Konečný krok	Parametry	Přesnost (%)	Úplnost (%)
exp1	3	CUT_COVERAGE_ALLELES = 90 CLOSE_DISTANCE = 100	65	84
	3	CUT_COVERAGE_ALLELES = 70 CLOSE_DISTANCE = 100	49	90
exp2	3	CUT_COVERAGE_ALLELES = 70 CLOSE_DISTANCE = 100	48	91
	4	CUT_COVERAGE_ALLELES = 70 CLOSE_DISTANCE = 100	78	87
exp3	4	CUT_COVERAGE_ALLELES = 70 CLOSE_DISTANCE = 100 CLUSTER_DISTANCE = 5	78	87
	4	CUT_COVERAGE_ALLELES = 70 CLOSE_DISTANCE = 100 CLUSTER_DISTANCE = 10	80	87
	4	CUT_COVERAGE_ALLELES = 70 CLOSE_DISTANCE = 100 CLUSTER_DISTANCE = 20	78	88
	4	CUT_COVERAGE_ALLELES = 70 CLOSE_DISTANCE = 100 CLUSTER_DISTANCE = 30	79	87

8 Závěr

Práce se zabývá návrhem nástroje pro automatickou identifikaci KIR alel. Identifikace KIR alel je využitelná při transplantaci krvetvorných buněk, kdy se rozhoduje mezi více dárci shodných v HLA znacích. Vybrání dárce s vhodnějšími KIR geny snižuje riziko relapsu (návratu nemoci). V teoretické části byly popsány a rozbrány geny, především non-HLA gen Killer-cell immunoglobulin-like receptor(KIR) a jejich vliv na transplantaci krvetvorných buněk. Dále byly představeny natural killer buňky a jejich funkce v rámci imunitního systému. Pro zjištění typizace HLA a KIR složí sekvenační metody, které byly shrnuty a popsány. Konkrétněji Sanger sekvenování a next-generation sequencing (NGS). V neposlední řadě byly analyzovány bioinformatické nástroje pro zpracování NGS dat se zaměřením na generátor syntetických readů ART a zarovnávač NGS readů Bowtie2. V realizační části byl navržen a implementován program v jazyce Python za pomocí knihovny pysam pro identifikaci KIR alel. Během práce bylo implementováno a vyzkoušeno několik přístupů, které byli v práci zhodnoceny.

Parametry ARTU byli nastaveny na základě dat z FN protože používají zrovna tenhle sekvenátor

Testování bylo prováděno na syntetických readech a následná validace byla provedena na datech z FN

Do budoucna by to chtělo co? navrhnout a implementovat přístup 4

9 Výkladový slovník pojmu a zkratek

WHO	World health organization, světová zdravotnická organizace
ČNRDD	Český národní registr dárců kostní dřeně
MHC	Major histocompatibility complex, genetický systém
HLA	Human leucocyte antigen, podskupina MHC
KIR	Killer imunoglobulin like-receptor, skupina genů
NK	Natural killer, buňka imunitního systému
DNA	Deoxyribonukleová kyselina; dvoušroubovice, která obsahuje páry bází C, G, A, T
RNA	Ribonuklové kyselina; obsahuje báze C, G, A, U; šablona přímo pro vytvoření proteinů; hlavní funkci zajištění překladu DNA do struktury proteinů (DNA -> mRNA -> rRNA -> tRNA -> RNA)
Báze	nukleové báze; A - Adenin, C - Cytosin, G - Guanin, T - Thymin
bp	base pair; jeden z párů A - T nebo C - G
kb	kilobase 1 kb = 1000 bp
ART	nástroj na vytváření syntetických readů
Bowtie	nástroj na zarovnání readů proti referenčním genům
SAM	Sequence Alignment/Map; Formát souboru na uložení zarovnání
BAM	Binární verze souboru SAM
Fenotyp	adwda
Genotyp	adawwd

TODO tímhle si nejsem moc jistá tak jsem to pochopila je to dobře?
DNA -> mRNA -> rRNA -> tRNA -> RNA

TODO co ty formáty souboru?

fenotyp tyhle kraviny Genotyp pro danou chromozomální oblast se pak u většiny lidí skládá ze dvou haplotypů). genom kompletní sekvence daného organismu

DNA (Deoxyribonukleová kyselina)

- dvoušroubovice, která obsahuje páry bází C, G, A, T

obojí obsahuje nukleotidy bází? Rozdíl mezi DNA a RNA DNA dvoušroubovice, která obsahuje páry bází - C G A T, kdežto RNA je již šablona přímo pro vytvoření proteinů takže jedna půlka šroubovice bez intronů. Hlavní funkcí RNA je zajištění překladu genetického kódu (DNA) do struktury proteinů nejdřív je DNA mRNA, rRNA tRNA RNA

Co znamená konec 3 a konec 5?

Literatura

- [1] *Chromosome* [online]. [cit. 2020/12/3]. Dostupné z:
<https://www.genome.gov/genetics-glossary/Chromosome>.
- [2] *DNA sequencing Fact Sheet* [online]. [cit. 2019/03/1]. Dostupné z:
<https://www.genome.gov/about-genomics/fact-sheets/DNA-Sequencing-Fact-Sheet>.
- [3] *S transplantací kostní dřeně stále častěji pomáhají příbuzní* [online]. Dostupné z: <https://ct24.ceskatelevize.cz/domaci/2527141-s-transplantaci-kostni-drene-stale-casteji-pomahaji-pribuzni>.
- [4] *Basic genetics* [online]. [cit. 2020/12/3]. Dostupné z:
<https://kintalk.org/genetics-101/>.
- [5] *Illumina* [online]. [cit. 2019/03/1]. Dostupné z:
<https://www.illumina.com/>.
- [6] *KIR genotypes* [online]. Dostupné z:
<http://www.allelefrequencies.net/kir6001a.asp>.
- [7] *What is the best NGS alignment software?* [online]. Dostupné z:
<https://www.ecseq.com/support/ngs/what-is-the-best/ngs-alignment-software>.
- [8] BARANWAL, A. – MEHRA, N. Major Histocompatibility Complex Class I Chain-Related A (MICA) Molecules: Relevance in Solid Organ Transplantation. *Frontiers in Immunology*. 02 2017, 8. doi: 10.3389/fimmu.2017.00182.
- [9] BERNAREGGI, D. – POUYANFARD, S. – KAUFMAN, D. S. Development of innate immune cells from human pluripotent steam cells. 2019. Dostupné z:
<https://www.sciencedirect.com/science/article/pii/S0301472X19300037?via%3Dihub>.
- [10] COOLEY, S. – WISDORF, D. J. – GUETHLEIN, L. A. Donor selection for natural killer cell receptor genes leads to superior survival after unrelated transplantation for acute myelogenous leukemia. 2010. Dostupné z:
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2953880/>.
- [11] ESCALONA, M. – ROCHA, S. – POSADA1, D. A comparison of tools for the simulation of genomic next-generation sequencing data. 2016. Dostupné z:
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5224698/#>.

- [12] FRYČOVÁ, M. Lze u pacientů s AML indikovaných k nepříbuzenské transplantaci provádět v klinické praxi výběr nepříbuzných dárců na základě KIR genotypů, 2016.
- [13] HERNYCHOVÁ, L. Receptory NK buněk. 2012.
- [14] HOUDOVÁ, L. et al. Optimal Donor Selection During Verification Process: Which Factors Are Worth Knowing? 2018. doi: 10.1016/j.transproceed.2018.08.043.
- [15] HUANG, W. et al. ART: a next-generation sequencing read simulator. 2012. Dostupné z: <https://academic.oup.com/bioinformatics/article/28/4/593/213322>.
- [16] J, R. et al. *Nomenclature* [online]. Nucleic Acids Research, 2015. [cit. 2019/10/1]. 43:D423-431. Dostupné z: <http://hla.alleles.org/misc/citing.html>.
- [17] KOLÍSKO, M. Moderní metody sekvenování DNA. 2017. Dostupné z: <https://ziva.avcr.cz/files/ziva/pdf/moderni-metody-sekvenovani-dna.pdf>.
- [18] LANGMEAD, B. [online]. [cit. 2019/03/1]. Dostupné z: <http://www.langmead-lab.org/>.
- [19] LANGMEAD, B. – SALZBERG, S. L. Fast gapped-read alignment with Bowtie 2. 2012. Dostupné z: https://www.researchgate.net/publication/221886241_Langmead_B_Salzberg_SL_Fast_gapped-read_alignment_with_Bowtie_2_Nat_Methods_9_357-359.
- [20] LANGMEAD, B. et al. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. 2009. Dostupné z: <https://genomebiology.biomedcentral.com/articles/10.1186/gb-2009-10-3-r25>.
- [21] MANIANGOU, B. et al. Killer Immunoglobulin-Like Receptor Allele Determination Using Next-Generation Sequencing Technology. *Frontiers in Immunology*. 2017, 8, s. 547. ISSN 1664-3224. doi: 10.3389/fimmu.2017.00547. Dostupné z: <https://www.frontiersin.org/article/10.3389/fimmu.2017.00547>.
- [22] MERZKER, M. L. Sequencing technologies-the next generation. 2013. doi: 10.1038/nrg2626.
- [23] MUDR. PAVEL JINDRA, P. D. *Imunopatologické a imunogenetické aspekty transplantací krvetvorných buněk a solidních orgánů*. PhD thesis, Universita Karlova v Praze, 2011.

- [24] PAPOUŠEK, I. Elektroforéza nukleových kyselin. 2017. Dostupné z: https://fvhe.vfu.cz/files/mbhp_2018_02.pdf.
- [25] PENDE, D. et al. *Killer Ig-Like Receptors (KIRs): Their Role in NK Cell Modulation and Developments Leading to Their Clinical Exploitation* [online]. 2019. Dostupné z: <https://www.frontiersin.org/articles/10.3389/fimmu.2019.01179/full>.
- [26] PENKA, M. – KOLEKTIV, E. T. *Hematologie a transfuzní lékařství II*. 2012. ISBN 978-80-247-3460-6.
- [27] ROBINSON, J. et al. IPD—the Immuno Polymorphism Database. 2013. Dostupné z: <https://www.ebi.ac.uk/ipd/index.html>.
- [28] ROBINSON, J. et al. The IPD and IMGT/HLA Database:allele variant databases. 2015. Dostupné z: <https://www.ebi.ac.uk/ipd/index.html>.
- [29] S.KANNANA, G. – ARIANEXYSQUINO-LOPEZ – A.LEED, D. Natural killer cells in malignant hematoloy: A primer for the non-immunologist. 2017. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S0268960X16300704>.
- [30] SMITH, D. T. *Encyklopédie lidského těla*. 2005. ISBN 80-7321-156-4.
- [31] THIELENS, A. – VIVIER, E. – ROMAGNÉ, F. NK cell MHC class I specific receptors (KIR): from biology to clinical intervention. *Current opinion in immunology*. 2012, 24 2, s. 239–45.
- [32] VODSLOŇ, F. Vyhodnocování podobnosti zdrojových textů, 2007.

A Uživatelská dokumentace

Program byl napsán a otestován za použití ART ve verzi MountRainier, Bowtie 2 ve verzi 2.4.1, Python ve verzi 3.8. Dále byla použita pytnovská knihovny pysam ve verzi 0.14.

Následující postupy jsou uvedeny pro operační systém Linux a pro jiné operační systémy se mohou lišit. Veškeré nastavení aplikace probíhá pomocí souboru *config.py*

Parametry configu:

- CREATE_READS - Značí zda má být spuštěn modul vytvoření syntetických readů. Očekávaný hodnota je True nebo False.
- ALIGN - Značí zda má být spuštěn modul pro zarovnání readů vzhledem k referenčním genům. Očekávaná hodnota je True nebo False.
- IDENTIFY - Značí zda má být spuštěna identifikace alel. Očekávaná hodnota je True nebo False.
- EXP1 - Značí zdá má být spuštěn experiment 1 pro identifikaci.
- EXP2 - Značí zdá má být spuštěn experiment 2 pro identifikaci.
- EXP3 - Značí zdá má být spuštěn experiment 3 pro identifikaci.
- REFERENCE_KIR_GENS_FILE - Soubor se všemi referenčními geny.
- GENOME_FOLDER - Označuje cestu složky do které jsou ukládány vytvořené genomy.
- GENOMES - Slovník, který definuje haplotypy podle obsahů genů. Na základě toho budou vytvořeny haplotypy.
- BOWTIE_HOME_DIRECTORY - Označuje cestu ke nástroji Bowtie.
- READS_FOLDER - Označuje složku do které budou ukládány ready. Případně z které budou načítány.
- BOWTIE_INDEX_FOLDER - Označuje složku do které budou ukládány indexy z Bowtie. Případně z které budou načítány.

- BOWTIE_BUILD_INDEX - Značí zda mají být vytvořeny Bowtie indexy. Pokud bude hodnota nastavena na False, musí být přítomny indexy z minulého běhu, jinak zarovnávání nebude fungovat. Očekávaná hodnota je True nebo False.
- BOWTIE_THREADS - Počet vláken na která má být Bowtie spuštěn.
- ALIGNMENT_FOLDER - Označuje složku do které budou ukládány zarovnané ready. Případně z které budou načítány.
- BAM_FOLDER - Označuje složku na uložení BAM souborů.
- RESULT_FOLDER - Označuje složku do které budou uloženy výsledky vyhodnocení.

A.1 Spuštění programu

Program je možné spustit z příkazové řádky pomocí příkazu *python run.py*. Podmínkou fungování tohoto postupu je, že je třeba se nacházet v umístění skriptu.

A.2 Doporučená adresářová struktura pro data

- data
 - alignments
 - bam
 - bowtie_index
 - haplotype
 - reads
 - result

A.3 Výstupy programu

V případě tvorby vlastních haplotypů s doporučenou adresářovou strukturou najdeme ve složce *haplotype* soubory s příponou *.fa*. Každý soubor obsahuje jeden haplotyp. Vytvořené ready se budou nacházet ve složce *reads*. Protože haplotypy jsou paired-end náleží každému haplotypu dva soubory s příponou *.fq*. Jeden s *1* na konci a druhý s *2* na konci. V případě zarovnání mohou být

výstupní soubory indexy ve složce *bowtie_index*. Kdy pro každý referenční gen je vytvořeno 6 souborů s příponou *bt2*. Výsledné zarování se pak nachází ve složce *alignments* ve formátu *.sam*. Vyhodnocení zarovnání se pak nachází ve složce *result* ve formátu *.txt*

A.4 Nastavení ART a jeho spuštění

tak jsem stáhla normálně nejnovější verzi z niehs.nih.gov a podle instrukcí co byli v souboru INSTAL dala

A.5 Nastavení ART a bowtie

pair end 250 dlouhy ready misto MSv3 použit MSv1 protože tak budou i data co dostanu -f 100 pokryti 100 -na značí že nemá vytvořit soubor zarovnání

A.5.1 pokus to nejak spustit

Takže když otebru hlavní README tak mi to říká že tam jsou READ ME pro jednotlivé verze sekvenatoru ..

pak se to musí skompilovat

```
./configure --prefix=$HOME make make install
```

ted mě zajímá ta illumina tak podle README illumina tak můžu vložit do složky examples a tam pustit skript run_test_examples_illumina.sh , tak tam jsou 4 příklady použití a pokud asi všechno dobré proběhne tak se mi zobrazí páár nových souborů ve složce examples..

FASTQ - *.fq data file s ready. pro paired-end simulator *1.fq obsahuje data pro první ready a *2.fq různé druhé ready

tohle nějak funguje MSv3 tam musím dát aby to mohla dostat na délku ready 250 a znaci ze to je paired.. tak se má používat MSv1 *art_illumina-ssMSv3-sam-iamplicon_reference.fa-p-l250-f10-m300-s10-omoje_art_data* Tohle používá: *art_illumina-ssMSv1-sam-iamplicon_reference.fa-p-l250-f100-m300-s10-omoje_art_data*

A.6 Bowtie

a stáhla jsem to tady po kliknutí na bowtie binary release.

na strance 25.4 je řečeno o hledání tch nejlepších zarovnání a je tam možnost –best ale že je dvakrát nebo třikrát pomalejší než normální mod.. a

jde o to že najde první přijatelný a to označní kdežto při tom best prohledá co nejvíce a hledá to nejlepší i mezi téma přijatelnýma a to je pomalý.

tak jsem to stálka dala do složky a musela jsem teda nastavit proměnou prostředí export BT2_HOME=/home/kate/Dokumenty/FAV/Diplomka/existujicisw/bowtie 2.4.1 – linux – x86_64/ pak jsem pustila tohle: \$BT2_HOME/bowtie2-build \$BT2_HOME/example/reference/lambdavirus.falambda_virus a nakonec se mi vytvořili nějaký nový soubory lambda virus 1 atd.. v tom bowtie 2 adresáři

dělala jsem o podle tohohle webovky

indexy bowtie-build builds a Bowtie index from a set of DNA sequences. bowtie-build outputs a set of 6 files with suffixes .1.ebwt, .2.ebwt, .3.ebwt, .4.ebwt, .rev.1.ebwt, and .rev.2.ebwt. (If the total length of all the input sequences is greater than about 4 billion, then the index files will end in ebwtl instead of ebwt.) These files together constitute the index: they are all that is needed to align reads to that reference. The original sequence files are no longer used by Bowtie once the index is built.

A.7 Používané soubory

A.7.1 FASTQ

aln_start_pos označuje počáteční pozici v referenci sekvence, je vždy relativní vzhledem k vláknu referenční sekvence To znamená že aln_start_pos plus (10) vlákno je odlišné od aln_start_pos minus (-) vlákna... ???? WHAT????

ref_seq_aligned je zarovnaná oblast referenční sekvence, která může být plus vlákno nebo mínos vlákno referenční sekvence ref_seq_aligned je zarovnáný read, který je vždy ve stejné orientaci jako stejný read v odpovídajícím fastq suboru.

aln_start_pos is the alignment start position of reference sequence. aln_start_pos is always relative to the strand of reference sequence. That is, aln_start_pos 10 in the plus (+) strand is different from aln_start_pos 10 in the minus (-) strand.

ref_seq_aligned is the aligned region of reference sequence, which can be from plus strand or minus strand of the reference sequence. read_seq_aligned is the aligned sequence read, which always in the same orientation of the same read in the corresponding fastq file.

SAM je standardní formát pro NG sekvence ready zarování BED o tom tam nic není jen NOTE: both ALN and BED format files use 0-based coordinate system while SAM format uses 1-based coordinate system.

pak jsou tady 4 doporučené použití $art_illumina[options]-ss < sequencing_{system} >$
 $-sam - i < seqref file > -l < readlength > -f < foldcoverage >$
 $-o < outfile_{prefix} > art_illumina[options] - ss < sequencing_{system} >$
 $-sam - i < seqref file > -l < readlength > -c < num_{reads per sequence} >$
 $-o < outfile_{prefix} > art_illumina[options] - ss < sequencing_{system} >$
 $-sam - i < seqref file > -l < readlength > -f < foldcoverage >$
 $-m < mean_{fragsize} > -s < std_{fragsize} > -o < outfile_{prefix} >$
 $art_illumina[options] - ss < sequencing_{system} > -sam - i < seqref file >$
 $-l < readlength > -c < num_{reads per sequence} > -m < mean_{fragsize} >$
 $-s < std_{fragsize} > -o < outfile_{prefix} >$

A.7.2 FASTQ

Sekvenační přístroje produkují data ve formátu FASTQ takže i ART musí logicky generovat tenhle formát. Pokud jsou ready v páru tak je na konci .1 a druhý read z páru tam má .2 to jsem u těch svých přímo nenašla

ale máš teda tři druhy single end, paired-end a matepair.

FASTQ obsahuje obě základy sekvence ?? both sequence bases a kvality skore je to v následujícím formátu @read_id sequence read + base quality scores je kódovany by ascii code of a single character, kde je kvalita rovná score to ascii code character minus 33. chápou proč tam je to -33 protže když se podíváš do asci tabulky tak je tam od 33 první normální znak jinak jsou tam divný .. takže třeba otazník je v asci na 63 takže -33 takže má ohodnocení kvality 30 jen by mě teda zajímalo v jakém sme intervalu? - je 45 v asci a nevím jestli to je teda od 0 do 100? a teda nejvyšší číslo znamená nejkvalitnější a nejmenší míň kvalitní? Podle té diplomky to tak je že čím vyšší číslo tím kvalitnější a většinou je to od 0 do 40 jen zřídka to překročí hodnotu 60, když je tam 10 tak to znamená že jedna báze z deset je špatně.. když je tam 30 tak to znamená že jedna z 1000 je špatně. já tam mám třeba F a to je 70.

example: @refid-4028550-1 caacgccactcagcaatgatcggttattcacgt... +

ALN - zarovnání readů zase *1.aln pro první a *2.aln pro druhý soubor je rozdelen na hlavičku a body část obsahuje hlavičku a v té hlavičce je jakým příkazem byl soubor vygenerován a reference na sequnce id a jejich délku @CM tag pro příkaz a @SQ pro reference sequence Hlavička vždycky začíná s

HEADER EXAMPLE

B Testovací genomy

Test1	Test2	Test3	Test4
3DL3: 0030101, 0140201 2DS2: 0010104 2DL2: 0030105 2DL3: 0020101 2DL5B: 01301 2DS3: 0020102 (2x) 2DP1: 0030101, 0010203 2DL1: 0030203, 007 3DP1: 004 (2x) 2DL4: 0080104, 010 3DL1: 0150101 3DS1: 014 2DL5A: 00102 2DS5: 2DS1: 0020104 2DS4: 0010103 3DL2: 00501 (2x)	3DL3: 0090102, 019 2DS2: 2DL2: 0010101 2DL3: 0010111 2DL5B: 0020101 2DS3: 2DP1: 0020107, 0030102 2DL1: 0020102, 0030210 3DP1: 004, 01001 2DL4: 0080104 (2x) 3DL1: 0070101 3DS1: 078 2DL5A: 0050101 2DS5: 010 2DS1: 0020102 2DS4: 0010103 3DL2: 0020101, 00501	3DL3: 005, 0140201 2DS2: 2DL2: 2DL3: 0010101, 0020103 2DL5B: 2DS3: 2DP1: 0020108 (2x) 2DL1: 0040101, 008 3DP1: 0030102, 00902 2DL4: 0010306, 0050104 3DL1: 002, 0040101 3DS1: 2DL5A: 2DS5: 2DS1: 2DS4: 3DL2: 0010301, 008	3DL3: 0030104, 007 2DS2: 0010105 2DL2: 0030101 2DL3: 0010102 2DL5B: 2DS3: 2DP1: 008 2DL1: 007 3DP1: 007, 00902 2DL4: 0010307, 0080104 3DL1: 0150202 3DS1: 055 2DL5A: 2DS5: 007 2DS1: 0020101 2DS4: 0060101 3DL2: 0020101, 00903
Test5	Test6	Test7	Test8
3DL3: 0140202, 036 2DS2: 2DL2: 2DL3: 0010109, 006 2DL5B: 2DS3: 0010301 2DP1: 0030102, 009 2DL1: 0030208, 00303 3DP1: 001, 002 2DL4: 0010202 (2x) 3DL1: 0200101 3DS1: 0130102 2DL5A: 0010102 2DS5: 2DS1: 0020105 2DS4: 3DL2: 00202, 018	3DL3: 0090102, 0140203 2DS2: 0010111 2DL2: 0010105 2DL3: 0010102 2DL5B: 0080101 2DS3: 0010302 2DP1: 0020103, 010 2DL1: 0030203, 0040102 3DP1: 0030202, 0030402 2DL4: 0010303, 00901 3DL1: 0050102, 0250102 3DS1: 2DL5A: 2DS5: 2DS1: 2DS4: 0010104, 010 3DL2: 0010302, 01001	3DL3: 00802,0090103 2DS2: 2DL2: 2DL3: 0010103,0010108 2DL5B: 2DS3: 2DP1: 0020106, 004 2DL1: 0030204, 0030205 3DP1: 0030202 (2x) 2DL4: 0010201, 0010305 3DL1: 008, 0150203 3DS1: 2DL5A: 2DS5: 2DS1: 2DS4: 0010107, 0030104 3DL2: 0020105, 00901	3DL3: 0030103, 00601 2DS2: 0010103, 0010112 2DL2: 0010102, 0030101 2DL3: 2DL5B: 0070101 2DS3: 0020101 (2x), 0010302 2DP1: 0030102 2DL1: 00402 3DP1: 0030101, 005 2DL4: 00104, 0080104 3DL1: 3DS1: 0130104, 055 2DL5A: 0050102 (2x) 2DS5: 2DS1: 0020102, 0020105 2DS4: 3DL2: 0010102, 0070102
Test9	Test10	Test11	
3DL3: 0030103,00601 2DS2: 0010103, 0010112 2DL2: 0010102, 0030101 2DL3: 2DL5B: 0070101 2DS3: 0020101 (2x) 2DP1: 0030102 2DL1: 00402 3DP1: 0030101, 005 2DL4: 00104, 0080104 3DL1: 0150208 3DS1: 0130104 2DL5A: 01201 2DS5: 2DS1: 0020102 2DS4: 0040101 3DL2: 0010102, 0070102	3DL3: 0030101, 0140201 2DS2: 0010104 2DL2: 0030105 2DL3: 0020101 2DL5B: 01301 2DS3: 0020102 (2x) 2DP1: 0010203, 0030101 2DL1: 0030203, 007 3DP1: 004 (2x) 2DL4: 0080104, 010 3DL1: 0150101 3DS1: 014 2DL5A: 2DS5: 2DS1: 0020104 2DS4: 3DL2: 00501 (2x)	3DL3: 0030103, 00601 2DS2: 0010103, 0010112 2DL2: 0010102, 0030101 2DL3: 2DL5B: 2DS3: 2DP1: 0030102, 008 2DL1: 00402 (2x) 3DP1: 0030101, 005 2DL4: 00104, 0080104 3DL1: 3DS1: 0130104, 055 2DL5A: 0050102 (2x) 2DS5: 0020102, 0020103 2DS1: 0020102, 0020105 2DS4: 3DL2: 0010102, 0070102	

AMALA	BOB	COX	HO301
3DL3: 0040201, 00802 2DS2: 0010101 2DL2: 0030102 2DL3: 0010109 2DL5B: 2DS3: 2DP1: 0020108 2DL1: 0030201 3DP1: 007, 0090101 2DL4: 0010201, 0050106 3DL1: 0150201 3DS1: 0130101 2DL5A: 00102 2DS5: 0020101 2DS1: 0020106 2DS4: 0010101 3DL2: 0020105, 0070102	3DL3: 00101, 019 2DS2: 0010104 2DL2: 0030101 2DL3: 0020102 2DL5B: 2DS3: 2DP1: 0030101 2DL1: 0030210 3DP1: 002, 0030203 2DL4: 0010202, 0050101 3DL1: 002 3DS1: 0130105 2DL5A: 0010101 2DS5: 0020104 2DS1: 0020101 2DS4: 0010105 3DL2: 0020101, 0070102	3DL3: 00102, 0090101 2DS2: 2DL2: 2DL3: 0020101, 006 2DL5B: 2DS3: 2DP1: 0030102 (2x) 2DL1: 0020102 (2x) 3DP1: 005, 006 2DL4: 0050102, 00901 3DL1: 0050103 3DS1: 055 2DL5A: 2DS5: 0020102 2DS1: 0020105 2DS4: 010 3DL2: 0010301, 0070103	3DL3: 00102, 0090101 2DS2: 2DL2: 2DL3: 0020101, 006 2DL5B: 2DS3: 2DP1: 0030102 (2x) 2DL1: 0020102 (2x) 3DP1: 005, 006 2DL4: 00501, 00901 3DL1: 0050103 3DS1: 055 2DL5A: 2DS5: 0020102 2DS1: 0020105 2DS4: 010 3DL2: 0010301, 0070103
JVM	KAS011	OLGA	RSH
3DL3: 00801, 0140201 2DS2: 0010110 2DL2: 0030102 2DL3: 010 2DL5B: 2DS3: 2DP1: 004 2DL1: 0030203 3DP1: 001, 0030202 2DL4: 0010304, 0080101 3DL1: 0010104, 008 3DS1: 2DL5A: 2DS5: 2DS1: 2DS4: 0030103 (2x) 3DL2: 0010101, 018	3DL3: 0090101, 0140203 2DS2: 2DL2: 2DL3: 0020103 (2x) 2DL5B: 2DS3: 2DP1: 0020104, 0030101 2DL1: 0020101, 0030209 3DP1: 0030206, 009 2DL4: 0010301, 0050107 3DL1: 008 3DS1: 013011 2DL5A: 0010102 2DS5: 0020101 2DS1: 0020101 2DS4: 0030101 3DL2: 01001, 018	3DL3: 00201, 00202 2DS2: 2DL2: 2DL3: 0010105 (2x) 2DL5B: 2DS3: 2DP1: 0020105, 006 2DL1: 0030204 (2x) 3DP1: 0030201 (2x) 2DL4: 0050103, 00901 3DL1: 0010102, 0050101 3DS1: 0130107 2DL5A: 00103 2DS5: 0020103 2DS1: 0020101 2DS4: 010 3DL2: 0070101, 0070102	3DL3: 00202, 0040202 2DS2: 0010108 2DL2: 0030104 2DL3: 0010107 2DL5B: 004 2DS3: 2DP1: 0020110, 009 2DL1: 0030205, 01201 3DP1: 0030401, 008 2DL4: 0010307, 00901 3DL1: 0050101, 01701 3DS1: 2DL5A: 2DS5: 006 2DS1: 2DS4: 0060102 3DL2: 023, 056
WT51			
3DL3: 0090101, 036 2DS2: 0010103 2DL2: 0010107 2DL3: 006 2DL5B: 0020103 2DS3: 0020103, 0010302 2DP1: 0010202, 004 2DL1: 01201 (2x) 3DP1: 00303, 007 2DL4: 0050105, 0050103 3DL1: 3DS1: 0130102 (2x) 2DL5A: 0010103, 0050104 2DS5: 0020101 2DS1: 0020103 (2x) 2DS4: 3DL2: 00202, 00903			

C Detailní výsledky

C.1 Experiment1

63

Genotyp	Alel	Krok 1			Krok 2			Krok 3		
		Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc
amala	19 (0)	461	0 -	2 2DL5B 2DS3	113	2 2DL1*0030201 3DL1*0150201	0 -	23	4 3DP1*0090101 2DL4*0010201	0 -
bob	19 (0)	461	0 -	2 2DL5B 2DS3	100	2 3DL1*002 2DL1*0030210	0 -	28	2 -	0 -
cox	19 (2)	461	0 -	5 2DS3 2DL2 2DL5B 2DS2 2DL5A	73	1 2DL4*00901	0 -	20	3 3DL3*0090101 3DP1*006	0 -
ho301	24 (6)	461	0 -	5 2DL5A 2DS1 2DS5 3DS1 2DL3	80	0 -	0 -	20	4 3DL2*0020106 2DL1*00402 2DS3*0020103 3DL1*002	1 3DL1
jvm	17 (1)	461	0 -	6 2DL5B 2DS1 3DS1 2DL5A 2DS5 2DS3	76	2 2DL4*0080101 2DL1*0030203	0 -	26	2 -	0 -
kas011	20 (1)	461	0 -	4 2DL5B 2DS3 2DL2 2DS2	94	2 2DL4*0050107 3DL1*008	0 -	29	3 3DL2*01001	0 -
olga	21 (3)	461	0 -	4 2DL5B 2DS3 2DL2 2DS2	99	1 3DL1*0010102	0 -	24	2 2DP1*0020105	0 -
rsh	20 (0)	461	0 -	4 2DL5A 2DS1 2DS3 3DS1	98	3 2DL1*0030205 3DL1*01701 2DL4*0010307	0 -	25	4 2DP1*0020110	0 -
wt51	24 (2)	461	0 -	2 3DL1 2DS4	118	0 -	0 -	34	4 3DL3*0090101 2DL5A*0010103 2DS3*0020103 2DS1*0020103	1 2DS1
test1	23 (3)	461	0 -	1 2DS5	73	2 3DL1*0150101 2DL1*0030203	0 -	27	2 -	0 -
test2	21 (1)	461	0 -	2 2DS3 2DS2	82	1 3DL1*0070101	1 3DL1	23	3 2DP1*0020107 2DL1*0020102	1 -

test3	16 (1)	461	0 -	9	2DS1 2DL5A 2DL2 3DS1 2DL5B 2DS2 2DS4 2DS5 2DS3	69	3	3DL1*002 2DL1*008 3DL1*0040101	1	3DL1	17	4	2DL4*0010306	1 -
test4	18 (0)	461	0 -	3	2DL5B 2DS3 2DL5A 2DL5B 2DL2 2DS4 2DS5 2DS2	89	2	3DL1*0150202 2DL4*0010307	0 -	22	3	3DL2*0020101	0 -	
test5	19 (1)	461	0 -	5	2DL5A 2DS1 3DS1 2DS5 2DS2	83	1	3DL1*0200101	0 -	21	3	2DL3*0010109 2DL1*0030208	0 -	
test6	21 (0)	461	0 -	4	2DL5A 2DS1 3DS1 2DS5	100	3	2DL1*0030203 3DL1*0250102 2DL4*00901	0 -	29	4	3DP1*0030202	0 -	
test7	18 (1)	461	0 -	8	2DS1 3DS1 2DL2 2DL5B 2DS2 2DS3 2DL5A 2DS5	94	2	3DL1*0150203 2DL1*0030205	0 -	15	7	2DS4*0010107 2DL4*0010201 2DP1*0020106 3DL3*0090103 2DL3*0010103	0 -	
test8	24 (2)	461	0 -	4	3DL1 2DS5 2DS4 2DL3	100	0 -		0 -	26	2	2DS1*0020102 3DL3*0030103	0 -	
test9	22 (1)	461	0 -	2	2DS5 2DL3	99	2	3DL1*0150208 2DL4*00104	0 -	26	3	3DL2*0070102	0 -	
test10	21 (3)	461	0 -	3	2DL5A 2DS5 2DS4	69	2	3DL1*0150101 2DL1*0030203	0 -	24	2 -		0 -	
test11	24 (2)	461	0 -	5	2DL5B 3DL1 2DS3 2DS4 2DL3	110	0 -		0 -	28	0 -		0 -	

Tabulka C.2: Výsledky experimentu 1 na syntetických readech. Odřezány byly alely, které měli pokrytí menší než 90%. Za blízké byly považovány v případě kdy byla jejich vzdálenost mezi sebou menší než 100. Alel u genotypu značí počet v daném genotypu. Číslo v závorkách udává kolik alel je dvakrát v daném genotypu. V každém kroku zbývá alel je z kolik alel ještě zůstalo ve výběru, ztraceno určuje kolik alel má být v genotypu, ale algoritmus je vyřadil. Za tímto číslem jsou vypsané alely které byly ztraceny. V dalších krocích jsou vypsány alely bez těch které už byly ztraceny v předchozích krocích. Obdobně je to s geny navíc, které udávají počet a jaké geny již neobsahují žádnou z alel, která náleží do daného genomu.

Genotyp	Alel	Krok 1			Krok 2			Krok 3		
		Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc
amala	19 (0)	461	0 -	2	2DL5B 2DS3	193	0 -	0 -	41	2 2DL4*0010201 3DP1*0090101
bob	19 (0)	461	0 -	2	2DL5B 2DS3	207	0 -	1 -	43	1 3DL1*002 2 3DL1
cox	19 (2)	461	0 -	5	2DS3 2DL2 2DL5B 2DS2 2DL5A	156	0 -	0 -	24	2 3DL3*0090101 3DP1*006
ho301	24 (6)	461	0 -	5	2DL5A 2DS1 2DS5 3DS1 2DL3	151	0 -	1 -	29	3 3DL2*0020106 2DL1*00402 2DS3*0020103
jvm	17 (1)	461	0 -	6	2DL5B 2DS1 3DS1 2DL5A 2DS5 2DS3	177	0 -	0 -	40	0 -
kas011	20 (1)	461	0 -	4	2DL5B 2DS3 2DL2 2DS2	204	0 -	1 -	43	1 3DL2*01001
olga	21 (3)	461	0 -	4	2DL5B 2DS3 2DL2 2DS2	194	0 -	1 -	34	1 2DP1*0020105
rsh	20 (0)	461	0 -	4	2DL5A 2DS1 2DS3 3DS1	228	0 -	1 -	40	2 2DP1*0020110 2DL1*0030205
wt51	24 (2)	461	0 -	2	3DL1 2DS4	178	0 -	0 -	42	4 3DL3*0090101 2DL5A*0010103 2DS3*0020103 2DS1*0020103
test1	23 (3)	461	0 -	1	2DS5	152	1 3DL1*0150101	1 3DL1	33	1 -
test2	21 (1)	461	0 -	2	2DS3 2DS2	204	0 -	0 -	38	2 2DL1*0020102 2DP1*0020107

test3	16 (1)	461	0 -	9	2DS1 2DL5A 2DL2 3DS1 2DL5B 2DS2 2DS4 2DS5 2DS3	163	0 -	0 -	28	1	2DL4*0010306	0 -	
test4	18 (0)	461	0 -	3	2DL5B 2DS3 2DL5A 2DL5B 2DL2 2DS4 2DS5 2DS2	163	0 -	0 -	34	1	3DL2*0020101	0 -	
test5	19 (1)	461	0 -	5	2DL5A 2DS1 3DS1 2DS5 2DS2	187	0 -	1 -	31	2	2DL1*0030208 2DL3*0010109	1 -	
test6	21 (0)	461	0 -	4	2DL5A 2DS1 3DS1 2DS5	228	1 2DL1*0030203	1 -	46	2	3DP1*0030202	1 -	
test7	18 (1)	461	0 -	8	2DS1 3DS1 2DL2 2DL5B 2DS2 2DS3 2DL5A 2DS5	188	0 -	0 -	24	7	2DL4*0010201 3DL1*0150203 2DL3*0010103 2DP1*0020106 2DS4*0010107 3DL3*0090103 2DL1*0030205	0 -	
test8	24 (2)	461	0 -	4	3DL1 2DS5 2DS4 2DL3	138	0 -	0 -	33	2	3DL3*0030103 2DS1*0020102	0 -	
test9	22 (1)	461	0 -	2	2DS5 2DL3	145	1 3DL1*0150208	1 3DL1	37	2	3DL2*0070102	1 -	
test10	21 (3)	461	0 -	3	2DL5A 2DS5 2DS4	145	1 3DL1*0150101	2 3DL1	32	1 -		2 -	
test11	24 (2)	461	0 -	5	2DL5B 3DL1 2DS3 2DS4 2DL3	148	0 -	1 -	35	0 -		1 -	

Tabulka C.4: Výsledky experimentu 1 na syntetických readech. Odřezány byly alely, které měli pokrytí menší než 70%. Za blízké byly považovány v případě kdy byla jejich vzdálenost mezi sebou menší než 100. Alel u genotypu značí počet v daném genotypu. Číslo v závorkách udává kolik alel je dvakrát v daném genotypu. V každém kroku zbývá alel je z kolik alel ještě zůstalo ve výběru, ztraceno určuje kolik alel má být v genotypu, ale algoritmus je vyřadil. Za tímto číslem jsou vypsány alely které byly ztraceny. V dalších krocích jsou vypsány alely bez těch které už byly ztraceny v předchozích krocích. Obdobně je to s geny navíc, které udávají počet a jaké geny již neobsahují žádnou z alel, která náleží do daného genomu.

Genotyp	Alel	Krok 2			Krok 3		
		Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc
amala	19 (0)	344	2 2DL3*001 2DL2*00301	1 2DL5B	78	5 3DP1*00901 3DL2*0020105 3DL2*0070102	2 3DL2
bob	19 (0)	366	3 3DL3*01303 2DL3*00201 2DL2*00301	1 2DL5B	86	5 3DP1*00302 3DL2*0070102	1 -
cox	18 (0)	329	3 3DL3*00103 2DL3*007 2DL3*00201	1 2DL5B	68	4 3DP1*006	1 -
ho301	19 (2)	296	4 2DL5B*010 2DL2*00101 2DL2*00301 2DS2*002	4 2DL5B 2DL5A 2DS1 2DS5	65	6 2DL1*010 2DS3*00201	4 -
jvm	15 (0)	315	2 2DP1*005 2DL2*00301	1 2DP1	52	3 3DL3*00801	1 -
kas011	18 (0)	311	2 3DL3*01302 3DL2*019	1 2DL5B	70	5 2DP1*002 3DL3*00901 3DP1*00302	2 3DL3
olga	17 (0)	328	2 2DL3*00101 3DL3*00902	1 2DL5B	74	7 3DL2*00701 2DL4*011 3DL1*001 2DP1*006 3DP1*00302	3 3DL2 3DP1
rsh	15 (0)	361	0 -	5 2DS4 3DL2 2DL5A 2DS1 2DS3	73	1 2DP1*00201	5 -

wt51	12 (0)	297	1	3DL3*00103	4	3DP1 2DS3 2DL1 3DL2	66	2	2DL5A*00501	4	-
------	--------	-----	---	------------	---	------------------------------	----	---	-------------	---	---

Tabulka C.6: Výsledky experimentu 1 na reálných datech. Odřezány byly alely, které měli pokrytí menší než 70%. Za blízké byly považovány v případě kdy byla jejich vzdálenost mezi sebou menší než 100. Alel u genotypu značí počet v daném genotypu. Číslo v závorkách udává kolik alel je dvakrát v daném genotypu. V každém kroku zbývá alel je z kolik alel ještě zůstalo ve výběru, ztraceno určuje kolik alel má být v genotypu, ale algoritmus je vyřadil. Za tímto číslem jsou vypsány alely které byly ztraceny. V dalších krocích jsou vypsány alely bez těch které už byly ztraceny v předchozích krocích. Obdobně je to s geny navíc, které udávají počet a jaké geny již neobsahují žádnou z alel, která náleží do daného genomu.

69 C.2 Experiment2

Genotyp	Alez	Krok 2			Krok 3			Krok 4				
		Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc		
amala	19 (0)	193	0 -	0 -	38	2	2DL4*0010201 3DP1*0090101	0 -	18	3	3DL2*0020105	0 -
bob	19 (0)	207	0 -	1 2DL5B	43	2	3DL1*002 2DL4*0010202	2 3DL1	25	2 -	-	2 -
cox	19 (2)	156	0 -	0 -	24	1	3DL3*0090101	0 -	22	1 -	-	0 -
ho301	24 (6)	151	0 -	1 2DL5A	29	2	2DL1*00402 3DL2*0020106	1 -	18	3	2DS2*0010104	1 -
jvm	17 (1)	177	0 -	0 -	40	0 -	-	0 -	17	3	3DP1*0030202 2DL4*0080101 3DL3*00801	0 -
kas011	20 (1)	204	0 -	1 2DL5B	45	0 -	1 -	26	1	3DL3*0090101	1 -	
olga	21 (3)	194	0 -	1 2DL5B	36	1	2DP1*0020105	1 -	20	2	3DL1*0050101	1 -
rsh	20 (0)	228	0 -	1 2DL5A	44	2	2DP1*0020110 2DL1*0030205	1 -	20	4	3DL1*0050101 3DL3*0040202	1 -
wt51	24 (2)	178	0 -	0 -	43	2	2DS3*0020103 3DL3*0090101	0 -	24	2 -	-	0 -
test1	23 (3)	152	1 3DL1*0150101	1 3DL1	33	1 -	-	1 -	23	1 -	-	1 -

test2	21 (1)	204	0 -	0 -	38	2	2DL1*0020102 2DP1*0020107	0 -	23	3	3DL3*019	0 -
test3	16 (1)	163	0 -	0 -	28	2	2DL1*0040101 2DL4*0010306	0 -	15	2	-	0 -
test4	18 (0)	163	0 -	0 -	35	1	3DL2*0020101 2DL1*0030208	0 -	18	2	3DL3*007	0 -
test5	19 (1)	187	0 -	1 2DL5B	31	3	2DL3*0010109 2DP1*0030102	1 -	20	3	-	1 -
test6	21 (0)	228	1 2DL1*0030203	1 2DL5A	48	2	3DP1*0030202	1 -	26	4	2DP1*0020103 3DL3*0140203	1 -
							2DL3*0010103 2DL1*0030205 2DL4*0010201					
test7	18 (1)	188	0 -	0 -	24	7	3DL1*0150203 2DS4*0010107 2DP1*0020106 3DL3*0090103	0 -	13	8	3DL2*0020105	0 -
test8	24 (2)	138	0 -	0 -	35	1	2DS1*0020102	0 -	24	2	3DL2*0070102	0 -
test9	22 (1)	145	1 3DL1*0150208	1 3DL1	37	1	-	1 -	24	1	-	1 -
test10	21 (3)	145	1 3DL1*0150101	2 3DL1 2DL5A	33	1	-	2 -	22	1	-	2 -
test11	24 (2)	148	0 -	1 2DL5B	33	2	2DS5*0020103 2DS2*0010103	1 -	24	2	-	1 -

Tabulka C.8: Výsledky experimentu 2. Odřezány byly alely, které měli pokrytí menší než 70%. Za blízké byly považovány v případě kdy byla jejich vzdálenost mezi sebou menší než 100. Alel u genotypu značí počet v daném genotypu. Číslo v závorkách udává kolik alel je dvakrát v daném genotypu. V každém kroku zbývá alel je z kolik alel ještě zůstalo ve výběru, ztraceno určuje kolik alel má být v genotypu, ale algoritmus je vyřadil. Za tímto číslem jsou vypsáné alely které byly ztraceny. V dalších krocích jsou vypsány alely bez těch které už byly ztraceny v předchozích krocích. Obdobně je to s geny navíc, které udávají počet a jaké geny již neobsahují žádnou z alel, která naleží do daného genomu.

C.3 Experiment 3

|| Krok 2 || Krok 3 || Krok 4

Genotyp	Alel	Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc
amala	19 (0)	231	0 -	0 -	41	2	3DP1*0090101 2DL4*0010201	0 -	20	2 -
bob	19 (0)	228	0 -	1 2DL5B	41	1	3DL1*002	2 3DL1	24	2 2DL4*0050101
cox	19 (2)	165	0 -	0 -	25	2	3DP1*006 3DL3*0090101	0 -	20	2 -
ho301	24 (6)	162	0 -	1 2DL5A	29	2	2DL1*00402 3DL2*0020106	1 -	18	3 2DS2*0010104
jvm	17 (1)	199	0 -	0 -	38	0 -		0 -	17	3 2DL4*0080101 3DL3*00801 3DP1*0030202
kas011	20 (1)	213	0 -	1 2DL5B	43	1	2DP1*0020104	1 -	24	2 3DL3*0090101
olga	21 (3)	205	0 -	1 2DL5B	34	2	3DL2*0070102 2DP1*0020105	1 -	19	2 -
rsh	20 (0)	250	0 -	1 2DL5A	42	2	2DP1*0020110 2DL1*0030205	1 -	20	4 3DL1*0050101 3DL3*0040202
wt51	24 (2)	208	0 -	0 -	45	3	2DL5A*0010103 2DS3*0020103 3DL3*0090101	0 -	26	3 -
test1	23 (3)	174	0 -	0 -	35	0 -		0 -	24	0 -
test2	21 (1)	208	0 -	0 -	37	2	2DL1*0020102 2DP1*0020107	0 -	23	3 3DL3*019
test3	16 (1)	181	0 -	0 -	28	1	2DL4*0010306	0 -	15	2 2DL1*0040101
test4	18 (0)	183	0 -	0 -	35	2	2DS1*0020101 3DL2*0020101	1 2DS1	17	2 -
test5	19 (1)	198	0 -	1 2DL5B	32	3	2DL1*0030208 2DL3*0010109 2DP1*0030102	1 -	20	3 -
test6	21 (0)	279	0 -	1 2DL5A	51	1	3DP1*0030202	1 -	27	3 3DL3*0140203 2DP1*0020103
test7	18 (1)	193	0 -		23	7	3DL3*0090103 2DS4*0010107 2DL1*0030205 2DL3*0010103 2DP1*0020106 2DL4*0010201 3DL1*0150203	0 -	13	8 3DL2*0020105
test8	24 (2)	150	0 -	0 -	36	1	2DS1*0020102	0 -	25	2 3DL2*0070102
test9	22 (1)	174	0 -	0 -	37	0 -		0 -	22	1 3DL2*0070102
test10	21 (3)	162	1 3DL1*0150101	2 3DL1 2DL5A	34	1 -		2 -	23	1 -
test11	24 (2)	148	0 -	1 2DL5B	33	2	2DS5*0020103 2DS2*0010103	1 -	24	2 -

Tabulka C.10: Výsledky experimentu 3. Odřezány byly alely, které měli pokrytí menší než 70%. Za blízké byly považovány v případě kdy byla jejich vzdálenost mezi sebou menší než 100. Shluky vytvářeli alely, které od sebe měli vzdálenost maximálně 5. Alel u genotypu značí počet v daném genotypu. Číslo v závorkách udává kolik alel je dvakrát v daném genotypu. V každém kroku zbývá alel je z kolik alel ještě zůstalo ve výběru, ztraceno určuje kolik alel má být v genotypu, ale algoritmus je vyřadil. Za tímto číslem jsou vypsané alely které byly ztraceny. V dalších krocích jsou vypsány alely bez těch které už byly ztraceny v předchozích krocích. Obdobně je to s geny navíc, které udávají počet a jaké geny již neobsahují žádnou z alel, která náleží do daného genomu.

Genotyp	Alel	Krok 2			Krok 3			Krok 4		
		Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc
amala	19 (0)	237	0 -	0 -	37	2 3DP1*0090101 2DL4*0010201	0 -	16	3 3DL2*0020105	0 -
bob	19 (0)	242	0 -	1 2DL5B	43	0 -	1 -	24	1 2DL4*0050101	1 -
cox	19 (2)	175	0 -	0 -	23	2 3DP1*006 3DL3*0090101	0 -	19	3 3DL2*0070103	0 -
ho301	24 (6)	174	0 -	1 2DL5A	28	2 3DL2*0020106 2DL1*00402	1 -	18	3 2DS2*0010104	1 -
jvm	17 (1)	210	0 -	0 -	36	0 -	0 -	17	3 2DL4*0080101 3DL3*00801 3DP1*0030202	0 -
kas011	20 (1)	220	0 -	1 2DL5B	44	0 -	1 -	25	1 3DL3*0090101	1 -
olga	21 (3)	212	0 -	1 2DL5B	35	1 2DP1*0020105	1 -	20	2 3DL1*0050101	1 -
rsh	20 (0)	262	0 -	1 2DL5A	43	2 2DL1*0030205 2DP1*0020110	1 -	20	4 3DL3*0040202 3DL1*0050101	1 -
wt51	24 (2)	222	0 -	0 -	45	3 2DS3*0020103 2DL5A*0010103 3DL3*0090101	0 -	26	3 -	0 -
test1	23 (3)	188	0 -	0 -	34	0 -	0 -	23	1 3DL3*0030101	0 -
test2	21 (1)	209	0 -	0 -	37	2 2DL1*0020102 2DP1*0020107	0 -	24	3 3DL3*019	0 -
test3	16 (1)	191	0 -	0 -	28	1 2DL4*0010306	0 -	13	3 3DL1*002 2DL1*0040101	0 -
test4	18 (0)	192	0 -	0 -	39	1 3DL2*0020101 2DL1*0030208	0 -	17	2 3DL3*007	0 -
test5	19 (1)	208	0 -	1 2DL5B	32	3 2DP1*0030102 2DL3*0010109	1 -	19	3 -	1 -
test6	21 (0)	290	0 -	1 2DL5A	52	1 3DP1*0030202	1 -	25	3 2DP1*0020103 3DL3*0140203	1 -

3

test7	18 (1)	196	0 -	0 -	26	7	3DL1*0150203 2DL1*0030205 3DL3*0090103 2DP1*0020106 2DS4*0010107 2DL4*0010201 2DL3*0010103	0 -	14	8	3DL2*0020105	0 -
test8	24 (2)	158	0 -	0 -	36	1	2DS1*0020102	0 -	25	2	3DL2*0070102	0 -
test9	22 (1)	189	0 -	0 -	39	0	-	0 -	24	0	-	0 -
test10	21 (3)	174	0 -	1 2DL5A	31	0	-	1 -	21	0 -	-	1 -
test11	24 (2)	156	0 -	1 2DL5B	33	2	2DS5*0020103 2DS2*0010103	1 -	24	2 -	-	1 -

Tabulka C.12: Výsledky experimentu 3. Odřezány byly alely, které měli pokrytí menší než 70%. Za blízké byly považovány v případě kdy byla jejich vzdálenost mezi sebou menší než 100. Shluky vytvářeli alely, které od sebe měli vzdálenost maximálně 10. Alel u genotypu značí počet v daném genotypu. Číslo v závorkách udává kolik alel je dvakrát v daném genotypu. V každém kroku zbývá alel je z kolik alel ještě zůstalo ve výběru, ztraceno určuje kolik alel má být v genotypu, ale algoritmus je vyřadil. Za tímto číslem jsou vypsané alely které byly ztraceny. V dalších krocích jsou vypsány alely bez těch které už byly ztraceny v předchozích krocích. Obdobně je to s geny navíc, které udávají počet a jaké geny již neobsahují žádnou z alel, která naleží do daného genomu.

Genotyp	Alel	Krok 2			Krok 3			Krok 4			
		Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc	
amala	19 (0)	258	0 -	0 -	40	2	3DP1*0090101 2DL4*0010201	0 -	20	3 3DL2*0020105	0 -
bob	19 (0)	252	0 -	1 2DL5B	45	0 -	1 -	25	1 2DL4*0050101	1 -	
cox	19 (2)	177	0 -	0 -	23	2	3DP1*006 3DL3*0090101	0 -	21	2 -	0 -
ho301	24 (6)	180	0 -	1 2DL5A	30	2	2DL1*00402 3DL2*0020106	1 -	18	3 2DS2*0010104	1 -
jvm	17 (1)	227	0 -	0 -	36	0 -	0 -	17	3 2DL4*0080101 3DL3*00801 3DP1*0030202	0 -	
kas011	20 (1)	242	0 -	1 2DL5B	44	0 -	1 -	26	1 3DL3*0090101	1 -	
olga	21 (3)	213	0 -	1 2DL5B	38	1	2DP1*0020105	1 -	20	2 3DL1*0050101	1 -

→

rsh	20 (0)	270	0 -	1 2DL5A	44	2	2DL1*0030205 2DP1*0020110	1 -	20	4	3DL1*0050101 3DL3*0040202	1 -
wt51	24 (2)	223	0 -	0 -	47	2	3DL3*0090101 2DS3*0020103	0 -	29	2 -	0 -	0 -
test1	23 (3)	200	0 -	0 -	35	0 -	0 -	0 -	24	1	3DL3*0030101	0 -
test2	21 (1)	212	0 -	0 -	39	2	2DP1*0020107 2DL1*0020102	0 -	26	2 -	0 -	0 -
test3	16 (1)	199	0 -	0 -	30	1	2DL4*0010306	0 -	15	2	2DL1*0040101	0 -
test4	18 (0)	215	0 -	0 -	34	2	2DS1*0020101 3DL2*0020101	1 2DS1	17	2 -	1 -	1 -
test5	19 (1)	218	0 -	1 2DL5B	30	3	2DL1*0030208 2DL3*0010109 2DP1*0030102	1 -	19	3 -	1 -	1 -
test6	21 (0)	311	0 -	1 2DL5A	48	1	3DP1*0030202	1 -	26	2	2DP1*0020103	1 -
test7	18 (1)	201	0 -	0 -	24	7	2DP1*0020106 2DL1*0030205 3DL1*0150203 2DS4*0010107 3DL3*0090103 2DL4*0010201 2DL3*0010103	0 -	14	8	3DL2*0020105	0 -
test8	24 (2)	160	0 -	0 -	37	1	2DS1*0020102	0 -	24	2	3DL2*0070102	0 -
test9	22 (1)	204	0 -	0 -	38	0 -	0 -	0 -	23	0	0 -	0 -
test10	21 (3)	184	0 -	1 2DL5A	31	0 -	1 -	1 -	20	1	3DL3*0030101	1 -
test11	24 (2)	156	0 -	1 2DL5B	33	2	2DS2*0010103 2DS5*0020103	1 -	24	2 -	1 -	1 -

Tabulka C.14: Výsledky experimentu3. Odřezány byly alely, které měli pokrytí menší než 70%. Za blízké byly považovány v případě kdy byla jejich vzdálenost mezi sebou menší než 100. Shluky vytvářeli alely, které od sebe měli vzdálenost maximálně 20. Alel u genotypu značí počet v daném genotypu. Číslo v závorkách udává kolik alel je dvakrát v daném genotypu. V každém kroku zbývá alel je z kolik alel ještě zůstalo ve výběru, ztraceno určuje kolik alel má být v genotypu, ale algoritmus je vyřadil. Za tímto číslem jsou vypsané alely které byly ztraceny. V dalších krocích jsou vypsány alely bez těch které už byly ztraceny v předchozích krocích. Obdobně je to s geny navíc, které udávají počet a jaké geny již neobsahují žádnou z alel, která nalezní do daného genomu.

|| Krok 2 || Krok 3 || Krok 4

C7

Genotyp	Alel	Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc	Zbývá alel	Ztraceno	Geny navíc		
amala	19 (0)	281	0 -	0 -	39	2	2DL4*0010201 3DP1*0090101	0 -	19	3	3DL2*0020105	0 -
bob	19 (0)	276	0 -	1 2DL5B	41	0 -		1 -	23	2	3DL1*002 2DL4*0050101	2 3DL1
cox	19 (2)	223	0 -	0 -	26	2	3DP1*006 3DL3*0090101	0 -	22	3	3DL2*0070103	0 -
ho301	24 (6)	181	0 -	1 2DL5A	28	3	2DL2*0010103 3DL2*0020106 2DL1*00402	1 -	17	4	2DS2*0010104	1 -
jvm	17 (1)	268	0 -	0 -	36	0 -		0 -	17	3	3DP1*0030202 2DL4*0080101 3DL3*00801	0 -
kas011	20 (1)	259	0 -	1 2DL5B	43	0 -		1 -	26	1	3DL3*0090101	1 -
olga	21 (3)	249	0 -	1 2DL5B	35	1	2DP1*0020105	1 -	19	2	3DL1*0050101 3DL1*0050101	1 -
rsh	20 (0)	285	0 -	1 2DL5A	45	2	2DL1*0030205 2DP1*0020110	1 -	19	5	3DL3*0040202 3DL2*023	1 -
wt51	24 (2)	223	0 -	0 -	47	2	2DS3*0020103 3DL3*0090101	0 -	29	2	-	0 -
test1	23 (3)	227	0 -	0 -	35	0 -		0 -	24	1	3DL3*0030101	0 -
test2	21 (1)	218	0 -	0 -	39	2	2DP1*0020107 2DL1*0020102	0 -	23	3	3DL3*019	0 -
test3	16 (1)	238	0 -	0 -	28	1	2DL4*0010306	0 -	12	3	3DL1*002 2DL1*0040101	0 -
test4	18 (0)	218	0 -	0 -	35	2	3DL2*0020101 2DS1*0020101	1 2DS1	17	2	-	1 -
test5	19 (1)	219	0 -	1 2DL5B	32	3	2DP1*0030102 2DL3*0010109 2DL1*0030208	1 -	19	3	-	1 -
test6	21 (0)	337	0 -	1 2DL5A	47	1	3DP1*0030202	1 -	27	2	2DP1*0020103	1 -
test7	18 (1)	232	0 -	0 -			2DP1*0020106 2DL1*0030205 2DL3*0010103 3DL1*0150203 2DL4*0010201 2DS4*0010107 3DL3*0090103	0 -	14	8	3DL2*0020105	0 -
test8	24 (2)	163	0 -	0 -	36	1	2DS1*0020102	0 -	24	2	3DL2*0070102	0 -
test9	22 (1)	210	0 -	0 -	38	0 -		0 -	23	0 -	0 -	0 -
test10	21 (3)	216	0 -	1 2DL5A	32	0 -		1 -	20	1	3DL3*0030101	1 -
test11	24 (2)	159	0 -	1 2DL5B	35	0 -		1 -	26	0 -		1 -

Tabulka C.16: Výsledky experimentu 3. Odřezány byly alely, které měli pokrytí menší než 70%. Za blízké byly považovány v případě kdy byla jejich vzdálenost mezi sebou menší než 100. Shluky vytvářeli alely, které od sebe měli vzdálenost maximálně 30. Alel u genotypu značí počet v daném genotypu. Číslo v závorkách udává kolik alel je dvakrát v daném genotypu. V každém kroku zbývá alel je z kolik alel ještě zůstalo ve výběru, ztraceno určuje kolik alel má být v genotypu, ale algoritmus je vyřadil. Za tímto číslem jsou vypsané alely které byly ztraceny. V dalších krocích jsou vypsány alely bez těch které už byly ztraceny v předchozích krocích. Obdobně je to s geny navíc, které udávají počet a jaké geny již neobsahují žádnou z alel, která náleží do daného genomu.

Ukázky vzniklých clusterů

Níže jsou uvedeny největší clustery.

Při vzdálenosti 5 bylo vytvořeno kolem 224 clusterů.

28	17	15	11	11
2DL1*0030226	2DL1*0020112	2DL1*0040114	2DP1*0020107	3DL1*0020103
2DL1*0030219	2DL1*0020108	2DL1*00402	2DP1*0020105	3DL1*0020102
2DL1*0030230	2DL1*0020114	2DL1*0040107	2DP1*0020103	3DL1*0150215
2DL1*0030212	2DL1*0020113	2DL1*0040106	2DP1*0020109	3DL1*0020104
2DL1*0030229	2DL1*0020109	2DL1*0040109	2DP1*0020106	3DL1*0150214
2DL1*037	2DL1*0020110	2DL1*0040110	2DP1*0020108	3DL1*0020105
2DL1*0030210	2DL1*0020103	2DL1*0040101	2DP1*0020104	3DL1*0150216
2DL1*0030208	2DL1*0020102	2DL1*0040113	2DP1*0020102	3DL1*1190101
2DL1*0030211	2DL1*0020115	2DL1*0040105	2DP1*0020110	3DL1*1190102
2DL1*0030205	2DL1*0020106	2DL1*0040111	2DP1*0020101	3DL1*0150217
2DL1*025	2DL1*0020101	2DL1*0040104	2DP1*008	3DL1*0150218
2DL1*032N	2DL1*0020104	2DL1*0040103		
2DL1*0030214	2DL1*0020107	2DL1*0040108		
2DL1*0030221	2DL1*0020111	2DL1*007		
2DL1*0030216	2DL1*0020116	2DL1*0040115		
2DL1*0030213	2DL1*008			
2DL1*0030228	2DL1*0020105			
2DL1*0030218				
2DL1*0030209				
2DL1*0030204				
2DL1*0030227				
2DL1*0030215				
2DL1*0030217				
2DL1*0030203				
2DL1*0030224				
2DL1*0030202				
2DL1*0030231				
2DL1*0030223				
10	10	9	9	8
2DL4*043	2DL4*0080101	3DP1*0030203	2DL4*050	2DL4*0050105
2DL4*046	2DL4*0080107	3DP1*0030202	2DL4*0010202	2DL4*0050101
2DL4*0080203	2DL4*00803	3DP1*0030401	2DL4*00107	2DL4*049
2DL4*0080206	2DL4*0080402	3DP1*00303	2DL4*0010203	2DL4*0050106
2DL4*0080207	2DL4*0080401	3DP1*0030206	2DL4*054	2DL4*0050107
2DL4*052	2DL4*0080108	3DP1*0030201	2DL4*0010303	2DL4*0050102
2DL4*00805	2DL4*051	3DP1*008	2DL4*0010201	2DL4*0050103
2DL4*0080204	2DL4*053	3DP1*0030205	2DL4*042	2DL4*0050104
2DL4*0080208	2DL4*0080106	3DP1*0030204	2DL4*0010306	
2DL4*0080205	2DL4*0080105			

Při vzdálosti 10 bylo vytvořeno kolem 1771 clusterů.

33	17	15	14	13
2DL1*0030206	2DL1*0020115	2DL1*0040103	2DL5A*0050102	3DP1*0030202
2DL1*0030214	2DL1*0020103	2DL1*0040110	2DL5A*01201	3DP1*0030203
2DL1*0030203	2DL1*0020104	2DL1*0040109	2DL5A*01202	3DP1*0030206
2DL1*0030219	2DL1*0020105	2DL1*0040101	2DL5B*0020102	3DP1*0030201
2DL1*0030202	2DL1*008	2DL1*0040111	2DL5B*0020104	3DP1*0030204
2DL1*0030210	2DL1*0020111	2DL1*0040106	2DL5B*0020106	3DP1*00303
2DL1*032N	2DL1*0020112	2DL1*0040114	2DL5B*0020201	3DP1*008
2DL1*0030212	2DL1*0020106	2DL1*0040113	2DL5B*00603	3DP1*0030205
2DL1*0030208	2DL1*0020102	2DL1*0040105	2DL5B*0070102	3DP1*0030401
2DL1*0030211	2DL1*0020107	2DL1*0040108	2DL5B*0080102	3DP1*01001
2DL1*0030216	2DL1*0020116	2DL1*007	2DL5B*009	3DP1*0030402
2DL1*0030221	2DL1*0020108	2DL1*00402	2DL5B*01301	3DP1*005
2DL1*0030207	2DL1*0020109	2DL1*0040104	2DL5B*01302	3DP1*006
2DL1*0030222	2DL1*0020113	2DL1*0040107	2DL5B*01303	
2DL1*0030229	2DL1*0020101	2DL1*0040115		
2DL1*025	2DL1*0020114			
2DL1*0030218	2DL1*0020110			
2DL1*0030205				
2DL1*0030230				
2DL1*0030220				
2DL1*0030231				
2DL1*0030227				
2DL1*0030225				
2DL1*0030215				
2DL1*0030217				
2DL1*037				
2DL1*0030228				
2DL1*0030226				
2DL1*0030224				
2DL1*0030223				
2DL1*0030204				
2DL1*0030213				
2DL1*0030209				
11	11	11	11	10
2DL4*0010309	3DL1*0150103	3DL1*0150216	2DP1*0020104	2DL4*0080402
2DL4*0010305	3DL1*0150101	3DL1*0150218	2DP1*0020109	2DL4*0080105
2DL4*045	3DL1*0200101	3DL1*0150215	2DP1*0020103	2DL4*0080401
2DL4*0010307	3DL1*0150102	3DL1*0020102	2DP1*0020107	2DL4*00803
2DL4*042	3DL1*0150204	3DL1*0020103	2DP1*0020106	2DL4*053
2DL4*050	3DL1*077	3DL1*0020104	2DP1*0020101	2DL4*0080106
2DL4*0010306	3DL1*0200102	3DL1*1190101	2DP1*0020105	2DL4*0080107
2DL4*0010203	3DL1*0250103	3DL1*1190102	2DP1*0020102	2DL4*051
2DL4*0010202	3DL1*0150208	3DL1*0020105	2DP1*008	2DL4*0080101
2DL4*0010303	3DL1*0150210	3DL1*0150217	2DP1*0020108	2DL4*0080108
2DL4*00107	3DL1*0150205	3DL1*0150214	2DP1*0020110	

Při vzdálenosti 30 bylo vytvořeno 122 clusterů. Zde je uveden jen jeden ze zajímavějších jelikož obsahuje jak alely genu 2DL5A tak 2DL5B.

14
2DL5A*01201
2DL5B*0080102
2DL5A*01202
2DL5B*00603
2DL5B*0070102
2DL5B*0020104
2DL5B*01303
2DL5B*01301
2DL5B*0020201
2DL5B*0020106
2DL5B*0020102
2DL5A*0050102
2DL5B*01302
2DL5B*009

D Vyhodit

TODO vyhodit? Doposud nebylo zavedeno konkrétní pravidlo na pojmenování KIR genů. Avšak bylo navrženo, aby každý KIR haplotyp byl označen "KH –" následovaným trojmístným číslem, které bude označovat konkrétní haplotyp. Bylo by tak možné pojmenovat 999 haplotypů. Dále by bylo označeno do které ze dvou skupin A či B haplotyp patří. Skupina B musí obsahovat alespoň jeden z genů KIR2DL5, KIR2DS1, KIR2DS2, KIR2DS3, KIR2DS5 a KIR3DS1. Naopak skupina A neobsahuje ani jeden z těchto genů. Z tohoto pravidla je patrné, že haplotypy B mají vždy více aktivačních KIR než haplotypy A. Za trojmístným číslem by tedy dále bylo písmeno A nebo B. Nakonec by byl připojen 17-ti místný binární kód, který by označoval přítomnost "1" či absenci "0" genu. Pořadí genů by odpovídalo pořadí v genomu od centrometrické části k telemetrické části. [28]

Výsledné pojmenování by mohlo vypadat následovně:

$$KH - 001A - 11100010011011011 \quad (\text{D.0.1})$$

Analogicky by bylo pojmenování KIR genotypů. Označení by začínalo "KG –" následovaným čtyřmístným unikátním číslem označující daný genotyp. V neposlední řadě by opět byl 17-ti místní binární kód označující přítomnost či absenci KIR genu.

TODO možná informaci o B-content score za tohle, proč u více shodných se řeší KIR, že se vybírají B haplotypový dárci a že v poslední době se řeší, zda kromě haplotypu nemají vliv konkrétní alelické varianty KIR genů (to je to, proč vy to řešíte v diplomce) - s tímhle ještě moc nevím co budu dělat.

454 je sekvenování při kterém se zachycuje vyzářené světlo na základě toho pokud se báze přidala do řetězce či nikoliv. Jeho dominantní chybou je tedy nesprávné určení počtu přidaných bází. Pravděpodobnost chyby roste s frekvencí dlouhých úseků obsahující stejnou bázi. Proto ART modeluje rozdělení chyb na základě délky úseku obsahující stejnou bázi spolu s Markovovy řetězci.

SOLId je založené na označení čtyř barev pro 16 různých skupin bází. Pro paired-end read simulaci délky fragmentu je použito Gausovské rozdělení. Rozdělení chyb je založena na empirické znalosti získané z readů genero-

vaných Applied Biosystémem. ART zároveň nabází nastavené chybovosti základě lineárního měřítka.

Podle [15] je dostupných několik nástojů pro simulaci NGS dat (Wgsim, MetaSim, SimSeq, FlowSim), které fungují dobře pro sekvenátory pro které byli určeny, ale žádný z nich se nedokázal vypořádat se všemi nejvíce používanými. Jejich slabinou je především v generovaní chyb na základě jednotlivých módů konkrétního sekvenátoru. Nejčastější chyby jsou substituční a vložení či smazání (INDEL - insert-deletion). ART obsahuje technologické profily chyb a navíc mu může použít i uživatelský profil chyb. Profily které obsahují délky readů a chyby byly získány z datasetu skutečných sekvenovaných dat.

TODO možná napsat co znamenají konkrétní chyby

TODO Proč? No protože přesně ví co tam dávají za data, protože mu podšoupnou ten referenční genom a tak pak můžou dobře sledovat co ten zarovnávač s tím dělá. A proč je to o tolik výhodnější než když by měli nějaké realnej dataset? Možná že si tam můžou ty chyby navolit tak jak se jim hodí? Jako bude v tom míň chyb, ale stejně.

Počítání jen pokrytí za použití pysam.coverage či co nevím jestli se nějak víc štourat v těch bowtie nastavení.. u artu to nemá smysl protože tam se hlavně musím držet toho aby byli co nejvíce podobní těm z nemocnice bowtie se to snaží někam dát za každou cenu.

zkoušela jsem v pythonu přes pysam.depth, zkusit nějaký odhad na alely.